EMC® Solutions Enabler Symmetrix® SRDF Family CLI

Version 7.6

Product Guide

EMC²

Copyright © 2003 - 2013 EMC Corporation. All rights reserved. Published in the USA.

Published October, 2012

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date regulatory document for your product line, go to the technical documentation and advisories section on the EMC online support website.

CONTENTS

Preface

Part 1 Concepts and Procedures

Chapter 1 Getting Started

Introduction 22
Licensing requirements 23
SRDF and RecoverPoint 23
SRDF documentation 24
SYMCLI 25
SYMCLI commands for SRDF 26
Understanding SRDF pair states and links 28
SRDF pair states 29
Invalid tracks in SRDF pairs 30
SRDF device and link state combinations 30
Pinging Symmetrix arrays through SRDF links 31
Understanding and setting SRDF modes of operation 32
Synchronous 32
Semi-synchronous 33
Adaptive copy 33
Asynchronous 35
Domino effect 36
Domino effect on 36
Verifying SRDF modes and pair states 37
Verifying SRDF modes 37
Verifying SRDF pair states 39
Verifying SRDF modes and pair states simultaneously 40
Operational considerations 41
Symmetrix array access rights 41
SRDF operations and copy sessions 41
Migrating data from R1 to a larger R2 device 41
Preventing synchronization actions 42
Device external locks 42
Enabling SRDF software and hardware compression 42
SRDF/A and the consistency exempt option 43
Mixed-mode workloads on an SRDF director 43
FAST VP SRDF coordination 44

Chapter 2 SRDF Control Operations

Device groups, device files and composite groups 48
Device groups 48
Device files 48
Composite groups 49
Querying and verifying 49
Performing SRDF control operations 50
Command option descriptions 52
Activate and deactivate SRDF/A Delta Set Extension 56

Activate and deactivate SRDF/A write pacing 57
Cleanup operation for SRDF/A 59
Enable and disable consistency protection for SRDF/A devices 59
Establish (full) 59
Establish (incremental) 61
Failback 63
Failover 66
Invalidate R1 tracks 67
Invalidate R2 tracks 68
Make R1 not ready 69
Make R1 ready 69
Make R2 not ready 70
Make R2 ready 70
Merge track tables 70
Move one-half of an SRDF pair 71
Move SRDF device pairs 71
Read/write disable target device 71
Refresh R1 72
Refresh R2 73
Restore (full) 73
Restore (incremental) 75
Resume I/O on links 76
Split 77
Suspend I/O on links 79
Swap one-half of an SRDF pair 80
Swap SRDF pairs 80
Update R1 mirror 81
Write disable R1 83
Write disable R2 84
Write enable R1 84
Write enable R2 84
Displaying SRDF information 86
Finding Symmetrix arrays attached to hosts 86
Viewing command results 86
Viewing time estimate for pair synchronization 86
Showing SRDF device settings 86
Listing SRDF group settings 87
Listing directors with a specific group number 87
Listing composite groups 87
Listing SRDF/Star composite groups 87
Listing SRDF devices 88

Chapter 3 Dynamic SRDF Operations

Dynamic SRDF group operations 92
Adding dynamic groups 93
Modifying dynamic groups 95
Renaming the label of a dynamic group 96
Removing dynamic groups 96
Removing a dynamic group from one side of an SRDF configuration 96
Dynamically changing Link Domino and Autolink Recovery modes 97
Specifying a link limbo for a dynamic group 97
Setting SRDF group attributes 98
Dynamic SRDF device pair operations 99
Creating a device file 99

Creating dynamic SRDF device pairs 100
Blocking createpair when R2 is larger than R1 104
Deleting dynamic SRDF pairs 104
Deleting one-half of a pair 105
Displaying dynamic SRDF devices 106
Grouping dynamic pairs with a device file 106
Deleting SRDF dynamic pairs 106
Moving dynamic SRDF pairs 107
Moving one-half of an SRDF pair 108
Cascaded SRDF device moves 108
SRDF mode after a movepair 108
Issuing a dynamic R1/R2 swap 108
Issuing a dynamic half-swap 109
Swapping cascaded SRDF devices 109
Displaying SRDF swap-capable devices 109
Swapping SRDF devices 110
Issuing dynamic failover establish 111
Issuing dynamic failover restore 112

Chapter 4 SRDF/Asynchronous Mode

Overview 114
SRDF/A benefits and features 114
SRDF/A restrictions 115
SRDF/Asynchronous operations 116
SRDF/A session monitoring 116
SRDF/A ordered-write processing 117
Setting SRDF/Asynchronous mode 118
Setting SRDF/A attributes 119
Checking for R1 invalid tracks 119
Using the consistency exempt option 120
Displaying SRDF/A session status 122
Listing SRDF/A device information 124
Using the immediate option 126
Using BCVs to preserve the R2 SRDF/A data copy 126
Confirming the R2 data copy 126
Transitioning to synchronous mode 127
Enabling consistency for concurrent SRDF/A devices 128
Managing SRDF/A Delta Set Extension pools 128
Using transmit idle 133
Using SRDF/A write pacing 133

Chapter 5 Concurrent SRDF Operations

Overview 142
Using concurrent R2 devices 142
Operating both mirrors in SRDF/S mode 143
Operating both mirrors in SRDF/A mode 143
Concurrent SRDF operations 144
Applicable pair states for concurrent SRDF operations 145
Establishing concurrent SRDF devices 145
Viewing concurrent SRDF devices 145
Splitting concurrent SRDF devices 145
Enabling consistency protection 146
Restoring R1 in a concurrent SRDF configuration 146

Restoring both R1 and R2 in a concurrent SRDF configuration 147

Chapter 6 Cascaded SRDF Operations

Overview 150
Benefits 151
Restrictions 151
Setting up cascaded SRDF 152
Applicable pair states for cascaded SRDF operations 153
Cascaded SRDF rules 153
RDF21 SRDF groups 154
Listing cascaded SRDF devices 154
SRDF mirror-based controls of an R21 device 155
Hop 2 controls 156
Querying hop 2 information 157
Listing R21 devices 160
SRDF/Extended Distance Protection 161
SRDF/EDP rules 163
Setting up cascaded SRDF/EDP 163
Managing a diskless cascaded environment 164
SRDF control and set operations 165
Dynamic SRDF controls 166
SRDF query support 166
Creating diskless devices 166
Adding an SRDF mirror 167
Restrictions 167
Viewing diskless devices 168
Restarting a diskless SRDF environment 170

Chapter 7 TimeFinder and SRDF

TimeFinder consistent splits across SRDF 172
Enginuity Consistency Assist 172
Multi-hop operations 173
Multi-hop SRDF sites 173
System-wide device groups 174
Examples of system-wide splits 175
Targeting commands to various multi-hop devices and links 176

Chapter 8 SRDF/Automated Replication

Overview 180
Restrictions 180
SRDF/Automated Replication operations 181
Single-hop data copies 181
Multi-hop data copies 185
Concurrent BCVs with SRDF/AR 187
Replication cycle patterns 188
Cycle time and invalid track statistics 189
Replication log entries 190
Clustered SRDF/AR environments 190
Setting replication retry and sleep times 191
Setting the symreplicate file parameters 192
Locked devices 197

Chapter 9 SRDF Consistency Group Operations

Overview 200
Consistency protection using the SRDF daemon 200
Redundant consistency protection 201
SRDF consistency group operations 202
Creating an SRDF consistency group 203
Creating composite groups from various sources 204
Enabling and disabling SRDF consistency protection 205
Enabling SRDF consistency protection for concurrent SRDF devices 208
Enabling SRDF consistency protection for cascaded SRDF devices 210
Checking if device pairs are enabled for consistency protection 210
Blocking symcg enable on R2 side 211
Deleting an SRDF consistency group 211
Suspending SRDF consistency protection 211
Using the msc_cleanup command 212
Dynamic modification of SRDF consistency groups 213
SRDF daemon interaction 213
Command restrictions 214
Preparing the staging area 215
Dynamically adding devices 218
Dynamically removing devices 222
Recovering from a failed dynamic modify operation 224
SRDF consistency with a parallel database 225
SRDF consistency with BCV access at the target site 226

Chapter 10 SRDF Device Migration

Overview 230
Configuration requirements 230
R1 device migration 230
R2 device migration 233
R1 and R2 migration procedures 236
Before you begin 236
Restrictions and limitations 237
Sample R1 migration procedure 238
Sample R2 migration procedure 245
Applicable SRDF pair states for migration 247
Pair states for migrate -setup 248
Pair states for migrate -replace for first leg of concurrent SRDF 249
Pair states for migrate -replace for second leg of concurrent SRDF 251

Chapter 11 SRDF Automated Recovery

Overview 256
Requirements and strategies 257
The symrecover command 259
Launching SRDF Automated Recovery 259
The symrecover options file parameters 261

Part 2 Operational Examples

Chapter 12 Performing SRDF Control Operations

Example 1: Basic SRDF control operations 270

Example 2: Concurrent SRDF 292
Example 3: Creating a dynamic SRDF group 311
Example 4: Creating dynamic SRDF pairs 314
Example 5: Operating with SRDF asynchronous replication 321
Example 6: Using a composite group to control SRDF pairs 329
Example 7: Creating concurrent dynamic SRDF pairs 337
Example 8: Failing over cascaded SRDF 342

Chapter 13 Querying and Verifying with SRDF Commands

Example 1: Querying a device group 350
Example 2: Querying a composite group 367
Verifying if invalid tracks are owed to R1 from R2 373

Chapter 14 Implementing Consistency Protection

Example 1: Consistency protection in asynchronous mode 378
Example 2: Tripping a consistency group automatically 385
Example 3: Tripping a consistency group manually 388
Example 4: Creating a composite group from existing sources 393
Example 5: Consistency protection for concurrent SRDF 396
Example 6: Dynamic modification of SRDF consistency groups 404
RDF1 consistency group: Adding and removing devices 404
Cascaded RDF1 consistency group: Adding and removing devices 407
Concurrent RDF1 consistency group: Adding and removing devices 412
Example 7: Recovering from a failed dynamic add operation 417

Chapter 15 Performing SRDF/Automated Replication Operations

Example 1: SRDF/AR single-hop configuration 424
Example 2: SRDF/AR multi-hop configuration with BCVs at hop 2 433
Example 3: SRDF/AR single-hop configuration using a CG 435
Example 4: SRDF/AR multi-hop configuration using a CG 442
Example 5: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (single-hop configuration) 447
Example 6: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (multi-hop configuration) 451
Example 7: Restarting a replicate session when devices are locked 456

Appendix A TimeFinder Snap and Clone Reference

SRDF operations for TimeFinder/Snap and VP Snap sessions 460
SRDF operations on the R1 side 460
SRDF operations on the R2 side 464
SRDF operations for TimeFinder/Clone sessions 469
SRDF operations on the R1 side 469
SRDF operations on the R2 side 475
SRDF operations for Extent-level TimeFinder/Clone sessions 480
SRDF operations on the R1 side 480
SRDF operations on the R2 side 485
SRDF set operations for TimeFinder/Snap sessions 490
SRDF set operations on the R1 side 490
SRDF set operations on the R2 side 491
SRDF set operations for TimeFinder/Clone sessions 492
SRDF set operations on the R1 side 492
SRDF set operations on the R2 side 493

SRDF set operations for Extent-level TimeFinder/Clone sessions..... 494
SRDF set operations on the R1 side 494
SRDF set operations on the R2 side 495

Appendix B SRDF Pair State Reference

SRDF operations and applicable pair states..... 498
Cascaded SRDF operations and applicable pair states..... 502
Concurrent SRDF operations and applicable pair states 509
Concurrent R2 512
Consistency group operations and applicable pair states 514

Appendix C Rcopy State Rules Reference

Rcopy Session on the R1 side..... 518
R1 is part of an Rcopy PUSH 518
R1 is part of an Rcopy PULL 520
Rcopy Session on the R2 side..... 522
R2 is part of an Rcopy PUSH 522
R2 is part of an Rcopy PULL 525

FIGURES

	Title	Page
1	Basic SRDF configuration	22
2	RecoverPoint family	23
3	SRDF device and link states	28
4	Establishing an SRDF pair	61
5	Incremental establish of an SRDF pair	63
6	Failback of an SRDF device	65
7	Failover of an SRDF device	67
8	Restoring an SRDF device	74
9	Incrementally restoring an SRDF device	76
10	Splitting an SRDF pair	78
11	Update of SRDF device track tables	82
12	SRDF group topology in a switched SRDF solution	92
13	SRDF group topology in a point-to-point SRDF solution	93
14	SRDF/Asynchronous mode	117
15	Concurrent SRDF R1 configuration	142
16	Concurrent SRDF/S mirroring to recovery sites located near the workload site	143
17	Concurrent SRDF/A mirroring to recovery sites located far away	144
18	Restoring the source device in a concurrent configuration	146
19	Restoring the source device and mirror in a concurrent SRDF configuration	148
20	Cascaded SRDF configuration	150
21	Creating an R21 device	152
22	Determining cascaded SRDF pair state	155
23	Location of hop-2 devices	156
24	Creating an R21 diskless device	162
25	Adding a diskless SRDF mirror	167
26	ECA consistent split	173
27	Remote multi-hop SRDF configurations	178
28	Automated data copy path in single-hop SRDF systems	181
29	Automated data copy path in multi-hop SRDF systems	185
30	Concurrent BCV in a multi-hop configuration	188
31	Running redundant hosts to ensure consistency protection	202
32	Preparing the staging area for adding devices to the R1CG consistency group	216
33	R1CG consistency group after a dynamic modify add operation	216
34	Preparing the staging area for removing devices from the MyR1 CG	217
35	MyR1 CG after a dynamic modify remove operation	217
36	Adding a device to independently-enabled SRDF groups of a concurrent CG	220
37	Adding devices to independently-enabled SRDF groups of a cascaded CG	222
38	Using an SRDF consistency group with a parallel database configuration	225
39	Using an SRDF consistency group with BCVs at the target site	226
40	Configuration setup for an R1 migration	231
41	Establishing a concurrent SRDF relationship during an R1 migration	232
42	Replacing the source device during an R1 migration	233
43	Configuration setup for an R2 migration	234
44	Establishing a concurrent SRDF relationship during an R2 migration	235
45	Replacing a target device during an R2 migration	236
46	R1 migration example: Initial configuration	238
47	Concurrent SRDF relationship	241
48	Migrated R1 devices	243
49	R2 migration example: Initial configuration	245
50	Concurrent SRDF relationship	246

51 Migrated R2 devices 247
52 R1 migration: applicable R1/R2 pair states for migrate -setup 248
53 R2 migration: applicable R1/R2 pair states for migrate -setup 249
54 R1 migration: R11/R2 applicable pair states for migrate -replace (first leg) 250
55 R2 migration:R11/R2 applicable pair states for migrate -replace (first leg) 251
56 R1 migration: applicable R11/R2 pair states for migrate -replace (second leg) 252
57 R2 migration: applicable R11/R2 pair states for migrate -replace (second leg) 253
58 SRDF Recovery environment 256
59 Cascaded SRDF recovery environment 257
60 Initial cascaded SRDF configuration 342
61 SRDF mode is set to ACP Disk between SiteA and SiteB 347
62 Personality swap between SiteA and SiteB devices 348
63 Personality swap between SiteB and SiteC devices 348

TABLES

	Title	Page
1	SRDF documentation	25
2	SYMCLI commands for SRDF	26
3	SRDF pair states	29
4	Possible SRDF device and link state combinations	31
5	symrdf verify command options for verifying SRDF modes	38
6	Access rights required by a Symmetrix array	41
7	SRDF control operations	50
8	Basic options of the symrdf command	53
9	Command options for symrdf list	88
10	Device type combinations for creating SRDF pairs	100
11	SRDF device states before swap operation	111
12	SRDF modes allowed for cascaded configurations	151
13	SRDF modes allowed for cascaded SRDF/EDP	165
14	Remote multi-hop SRDF commands	176
15	Initial setups for cycle timing parameters	189
16	Supported consistency modes for concurrent mirrors	209
17	Consistency protection allowed for cascaded SRDF pairs	210
18	Consistency protection allowed for diskless cascaded SRDF pairs	210
19	Allowable device types for adding devices to an RDF1 CG	218
20	Allowable device types for adding devices to a concurrent RDF1 CG	219
21	Supported consistency modes for concurrent SRDF groups	219
22	Allowable device types for adding devices to a cascaded RDF1 CG	221
23	Supported consistency modes for cascaded hops	221
24	Allowable device types for removing devices from an RDF1 CG	223
25	Allowable device types for removing devices from a concurrent RDF1 CG	223
26	Allowable device types for removing devices from a cascaded RDF1 CG	224
27	SRDF migrate -setup control operation and applicable pair states	247
28	SRDF migrate -replace control operation and applicable pair states	249
29	SRDF migrate -replace control operation and applicable pair states	251
30	symrecover options file parameters	261
31	Allowable SRDF operations when R1 is the source of a TimeFinder/Snap or VP Snap	460
32	Allowable SRDF operations when R1 is the target of a TimeFinder/Snap or VP Snap	462
33	Allowable SRDF operations when R2 is the source of a TimeFinder/Snap or VP Snap	464
34	Allowable SRDF operations when R2 is the target of a TimeFinder/Snap or VP Snap	467
35	Allowable SRDF operations when R1 is the source of a TimeFinder/Clone	469
36	Allowable SRDF operations when R1 is the target of a TimeFinder/Clone	472
37	Allowable SRDF operations when R2 is the source of a TimeFinder/Clone	475
38	Allowable SRDF operations when R2 is the target of a TimeFinder/Clone	477
39	Allowable SRDF operations when R1 is the source of an Extent-level Clone	480
40	Allowable SRDF operations when the R1 is the target of an Extent-level Clone	482
41	Allowable SRDF operations when R2 is the source of an Extent-level Clone	485
42	Allowable SRDF operations when the R2 is the target of an Extent-level Clone	487
43	Allowable SRDF set operations when R1 is the source of a TimeFinder/Snap	490
44	Allowable SRDF set operations when R1 is the target of a TimeFinder/Snap	490
45	Allowable SRDF set operations when R2 is the source of a TimeFinder/Snap	491
46	Allowable SRDF set operations when R2 is the target of a TimeFinder/Snap	491
47	Allowable SRDF set operations when R1 is the source of a TimeFinder/Clone	492
48	Allowable SRDF set operations when R1 is the target of a TimeFinder/Clone	492
49	Allowable SRDF set operations when R2 is the source of a TimeFinder/Clone	493
50	Allowable SRDF set operations when R2 is the target of a TimeFinder/Clone	493

51 Allowable SRDF set operations when R1 is the source of an Extent-level Clone 494
52 Allowable SRDF set operations when R1 is the target of an Extent-level Clone..... 494
53 Allowable SRDF set operations when the R2 is the source of an Extent-level Clone 495
54 Allowable SRDF set operations when R2 is the target of an Extent-level Clone..... 495
55 SRDF control operations and applicable pair states 498
56 R1->R21 cascaded SRDF control operations and applicable pair states 502
57 R21->R2 cascaded SRDF control operations and applicable pair states 505
58 R1->R21 cascaded SRDF set operations and applicable pair states 508
59 R21->R2 Cascaded RDF Set Operations and Applicable Pair States..... 508
60 SRDF control operations and applicable states for concurrent R1 pairs..... 509
61 SRDF control operations and applicable states for concurrent R2 pairs..... 512
62 SRDF control operations and applicable pair states for devices in an SRDF/CG 514
63 Allowable SRDF operations with Rcopy PUSH session on the R1 518
64 Allowable SRDF operations with Rcopy PULL session on the R1 520
65 Allowable SRDF operations with Rcopy PUSH session on the R2 522
66 Allowable SRDF operations with Rcopy PULL session on the R2 525

PREFACE

As part of an effort to improve its product lines, EMC 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 EMC representative if a product does not function properly or does not function as described in this document.

Note: This document was accurate at publication time. New versions of this document might be released on EMC Online Support https://support.EMC.com. Check to ensure that you are using the latest version of this document.

Purpose

This document is part of the EMC Solutions Enabler documentation set, and describes how to use the Symmetrix Remote Data Facility (SRDF®).

Audience

This document is for advanced command-line users and script programmers to manage various types of control operations on Symmetrix arrays and devices using the SYMCLI commands of the Solutions Enabler software.

Related documentation

Solutions Enabler supports all VMAX family arrays. Refer to the EMC Symmetrix Product Guide for your specific VMAX series for details about which features are supported with your platform.

The following EMC publications provide additional information:

- *EMC Solutions Enabler Complete Documentation Set*
- *EMC Solutions Enabler Release Notes*
- *EMC Solutions Enabler Installation Guide*
- *EMC Symmetrix Security Configuration Guide*
- *EMC Solutions Enabler SYMCLI Command Reference Guide*
- *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
- *EMC Solutions Enabler Symmetrix Migration CLI Product Guide*
- *EMC Solutions Enabler TimeFinder Family CLI Product Guide*
- *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRM CLI Product Guide*
- EMC host connectivity guides for [your operating system]

For detailed interoperability information, please refer to E-Lab Interoperability Navigator, which can be reached at http://elabnavigator.EMC.com.

Note: Detailed man page descriptions of all SYMCLI commands, environment variables, option file parameters, and error codes are documented in the companion *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Typographical conventions

EMC uses the following type style conventions in this document:

Normal	Used in running (nonprocedural) text for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities • URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications
Bold	Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages Used in procedures for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • What the user specifically selects, clicks, presses, or types
Italic	Used in all text (including procedures) for: • Full titles of publications referenced in text • Emphasis, for example, a new term • Variables
`Courier`	Used for: • System output, such as an error message or script • URLs, complete paths, filenames, prompts, and syntax when shown outside of running text
`Courier bold`	Used for specific user input, such as commands
`Courier italic`	Used in procedures for: • Variables on the command line • User input variables
< >	Angle brackets enclose parameter or variable values supplied by the user
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections — the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained on EMC Online Support as described next.

Note: To open a service request through EMC Online Support, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.

Product information

For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to EMC Online Support (registration required) at:

https://support.EMC.com

Technical support

EMC offers a variety of support options.

Support by Product — EMC offers consolidated, product-specific information on the Web at:

https://support.EMC.com/products

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussion, relevant Customer Support Forum entries, and a link to EMC Live Chat.

EMC Live Chat — Open a Chat or instant message session with an EMC Support Engineer.

eLicensing support

To activate your entitlements and obtain your Symmetrix license files, visit the Service Center on http://support.EMC.com, as directed on your License Authorization Code (LAC) letter e-mailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact the EMC Customer Support Center.

If you are missing a LAC letter, or require further instructions on activating your licenses through EMC Online Support, contact EMC's worldwide Licensing team at licensing@emc.com or call:

- North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.
- EMEA: +353 (0) 21 4879862 and follow the voice prompts.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

techpubcomments@emc.com

PART 1

Concepts and Procedures

Part 1 of this product guide provides an introduction to Symmetrix Remote Data Facility (SRDF) and describes how to use the EMC Solutions Enabler Symmetrix Command Line Interface (SYMCLI) to perform SRDF operations on Symmetrix devices.

Concepts and procedures are described in the following chapters:

- Chapter 1, "Getting Started" - An introduction to SRDF and the SRDF component of the Solutions Enabler SYMCLI.

  Includes SRDF documents to read, considerations to be aware before invoking any SRDF operation, and a detailed overview of SRDF pair states.
- Chapter 2, "SRDF Control Operations" - SRDF control operations that enable you to establish, manage and view components comprising an SRDF configuration.
- Chapter 3, "Dynamic SRDF Operations" - Control operations to create and manage dynamic device pairs and groups.
- Chapter 4, "SRDF/Asynchronous Mode" - SRDF/Asynchronous mode and its control operations.
- Chapter 5, "Concurrent SRDF Operations" - How to create and perform operations on concurrent SRDF devices.
- Chapter 6, "Cascaded SRDF Operations" - How to set up a cascaded SRDF configuration and perform control operations on cascaded SRDF devices.
- Chapter 7, "TimeFinder and SRDF" - How to use the SYMCLI to utilize TimeFinder functionality.
- Chapter 8, "SRDF/Automated Replication" - How to perform automated consistent replication of data from standard devices and from RDF1 BCV devices over SRDF links to the remote SRDF pair.
- Chapter 9, "SRDF Consistency Group Operations" - How to create and maintain SRDF consistency groups using the SYMCLI.
- Chapter 10, "SRDF Device Migration" - How to migrate data from an existing R1 or R2 device to a new device, and then replace the existing device with the new device in an SRDF pair.
- Chapter 11, "SRDF Automated Recovery" - How to use the `symrecover` command to monitor the session state for device groups and composite groups.

CHAPTER 1
Getting Started

This chapter provides an introduction to SRDF and the SRDF component of the Solutions Enabler Symmetrix Command Line Interface (SYMCLI). Topics include:

- Introduction 22
- SRDF documentation 24
- SYMCLI 25
- Understanding SRDF pair states and links 28
- Understanding and setting SRDF modes of operation 32
- Verifying SRDF modes and pair states 37
- Operational considerations 41

Introduction

The EMC® Symmetrix® Remote Data Facility (SRDF®) family of products offers a range of Symmetrix-based disaster recovery, parallel processing, and data migration solutions.

SRDF disaster recovery solutions are based on active remote mirroring and dependent-write consistent copies of data maintained at one or more remote locations.

A dependent-write is a write operation that cannot be issued by an application until a prior, related write I/O operation completes. Dependent-write consistency ensures transactional consistency when the applications are restarted at the remote location.

SRDF configurations require at least two Symmetrix arrays; a *primary* system/array and at least one *secondary* system/array. Both sites can be located in the same room, in different buildings within the same campus, or hundreds to thousands of kilometers apart.

Note: The terms *SRDF* and *RDF* are used throughout this guide to refer to the Symmetrix Remote Data Facility. The terms *RDF group* and *RA* group are used interchangeably throughout this document as well as *RDF director* and *RDF adapter (RA)*.

Figure 1 shows a typical SRDF configuration in an open systems (left) and in a mainframe environment (right).

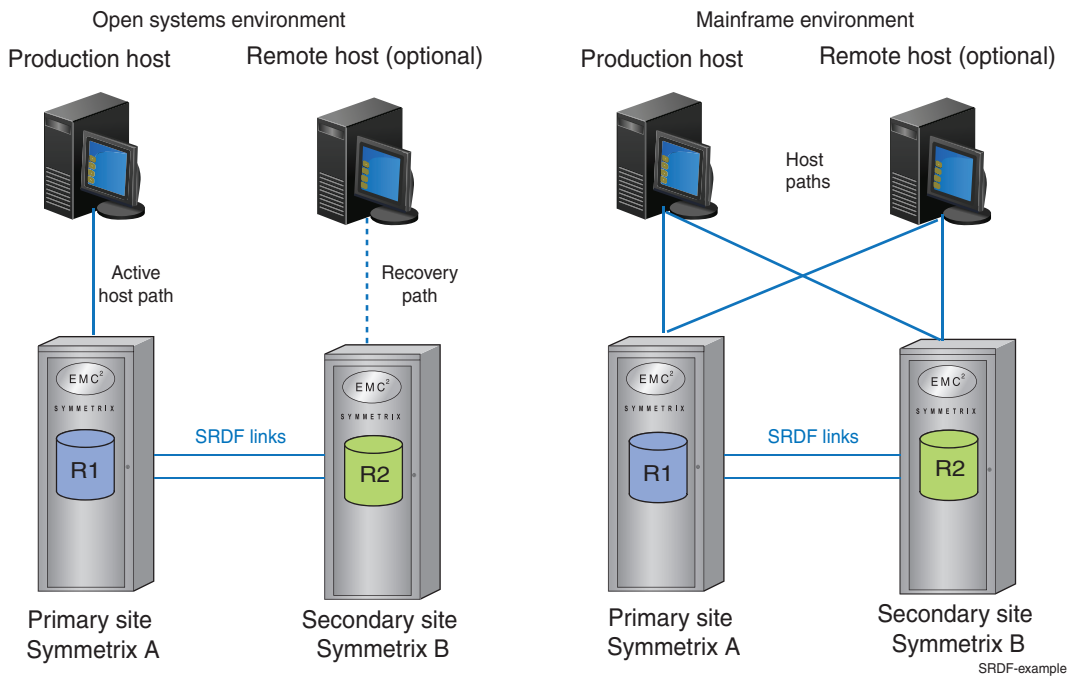


Figure 1 Basic SRDF configuration

In the open systems environment, the production host connects to Symmetrix A.

While the production host is in operation, it issues I/O (production data) to primary devices in Symmetrix A. SRDF remotely mirrors the production data to the secondary devices (R2) in Symmetrix B.

Symmetrix A and Symmetrix B are connected to each other over SRDF links.

In the mainframe environment, the production host and the remote host are connected to both Symmetrix A and Symmetrix B.

Mainframe hosts can leverage their operating system capabilities by configuring active, recovery, and alternate host paths to use the SRDF configuration. In this environment, SRDF supports both disaster recovery and parallel processing solutions.

Licensing requirements

The Enginuity™ Operating Environment must run on every Symmetrix array that participates in an SRDF solution.

You can configure different Symmetrix hardware models running different Enginuity versions within the same SRDF solution.

For all licensing information, see the *EMC Solutions Enabler Installation Guide*.

SRDF and RecoverPoint

EMC RecoverPoint provides comprehensive data protection by continuous replication (splitting) of host writes. With RecoverPoint, applications can be recovered to any point in time. Replicated writes can be written to local volumes to provide recovery from operational disasters, to remote volumes to provide recovery from site disasters, or both. The RecoverPoint familty includes 3 configurations; CDP, CRR, and CLR:

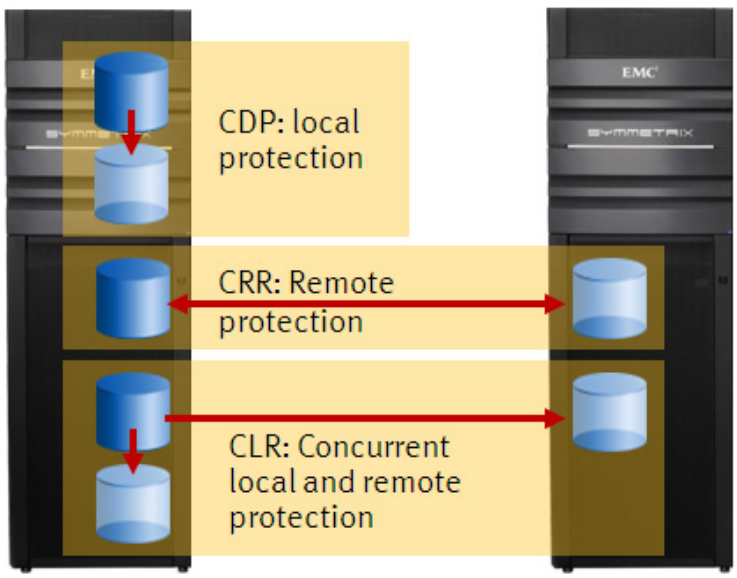


Figure 2 RecoverPoint family

Prior to Solution Enabler version 7.6:

- A device configured for use by RecoverPoint (also known as an RP tagged device) could not be configured as an SRDF device.
- A SRDF device could be tagged for use with RecoverPoint.

Starting in Soutions Enabler 7.6, a non-concurrent SRDF R1 device can be tagged for RecoverPoint; and a RecoverPoint tagged device is allowed to become SRDF R1 device. You can use the same source volume for both RDF (for disaster recovery) and RecoverPoint CDP (for local operational recovery).

This feature applies to:

- Both static and dynamic SRDF devices
- All SRDF modes; synchronous, asynchronous, and adaptive copy
- All VMAX platforms; VMAX 10K, VMAX 20K and VMAX 40K.

Note: This feature is supported with Enginuity 5876 Q2 2013 SR and above only.

A new flag, **-rp**, has been added to the syntax of the `symrdf` SYMCLI command.

The following operations will fail on a RecoverPoint tagged R1 device if the **-rp** flag is not specified:

- Restore, failback, or R1_update
- Resume
- Refresh
- Merge

R2 devices cannot be RecoverPoint tagged. The following actions on a RecoverPoint tagged R1 device fail:

- symrdf swap
- symrdf half_swap
- symrdf failover -establish|-restore

SRDF documentation

A complete list of SRDF documenation is listed in the Preface.

The EMC Online Support site (https://support.EMC.com) provides details about SRDF two-site interfamily connectivity options. On the home page of the EMC Online Support site, search with the keyword `SRDF Two-site Interfamily Connectivity`.

EMC recommends that you read the documents listed in Table 1 before invoking any SYMCLI SRDF control operation.

Table 1 SRDF documentation

For information on	See
Technical concepts and operations of the SRDF product family	*EMC Symmetrix Remote Data Facility (SRDF) for VMAX 40K, VMAX 20K/VMAX, DMX Series Product Guide* *EMC Symmetrix Remote Data Facility (SRDF) for VMAX 10K/VMAXe Series Product Guide*
How to configure and manage a Symmetrix configuration using the Solutions Enabler SYMCLI	*EMC Solutions Enabler Symmetrix Array Controls Product Guide*
Syntax of the Solutions Enabler SYMCLI commands	*EMC Solutions Enabler Symmetrix CLI Command Reference*
Securing your SRDF configuration	*EMC Solutions Enabler Security Configuration Guide*

SYMCLI

The Solutions Enabler SYMCLI is a set of commands that you can invoke from a host's command line, or using scripts.

Solutions Enabler SRDF extends the basic SYMCLI command set to include commands for performing control operations on SRDF devices. The SRDF SYMCLI commands run on many open systems servers and on mainframe servers that have Linux-based operating system partitions installed.

SYMCLI commands for SRDF

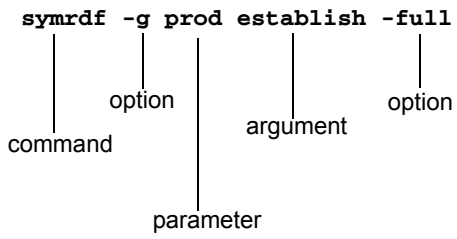
Table 2 lists the four main commands of the Solutions Enabler SRDF component used to establish, maintain and monitor SRDF configurations.

Table 2 SYMCLI commands for SRDF

Command	Description	For more information
symrdf	Performs the control operations on SRDF devices, such as: • Establishes (mirrors) an SRDF pair by initiating a data copy from the source (R1) side to the target (R2) side. This operation can be a full or incremental establish. • Restores remote mirroring. Initiates a data copy from the target (R2) side to the source (R1) side. This operation can be a full or incremental restore. • Splits an SRDF pair, which stops mirroring for the SRDF pairs in a device group. • Fails over and back from the source (R1) side to the target (R2) side, switching data processing to the target (R2) side. • Updates the source (R1) side after a failover, while the target (R2) side may still be operational to its local host(s). • Swaps the source (R1) and target (R2) destinations between the target and the source. • Creates, deletes, or swaps dynamic SRDF device pairs. • Performs dynamic SRDF group controls to add, modify, and remove dynamic groups. • Enables link domino locally or remotely when creating dynamic groups. • Enables auto link recovery locally or remotely when creating dynamic groups. • Enables/disables consistency for SRDF/A capable devices operating in asynchronous mode that are managed by a device group or file.	See page 47 and the `symrdf` man page.
symreplicate	Invokes a replicate session that generates automated recurrent, background copies of the standard data following a path across SRDF links and cascading BCVs. You can start, stop, and restart a replicate session. This command is used for SRDF/Automated Replication.	See page 179 and the `symreplicate` man page.
symstar	Uses concurrent and cascaded SRDF/Synchronous and SRDF/Asynchronous links to replicate source data synchronously to a nearby regional site and asynchronously to a distant remote site.	See the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide* and the `symstar` man page.
symrecover	Provides a session state monitoring tool that attempts to restart a group session if it enters the suspended or partitioned state.	See page 255 and the `symrecover` man page.

Command syntax

The following example shows the command syntax for initiating a full establish for the SRDF pairs in the `prod` device group.



Getting command help

Type `<command> -h` to display command line help for the specified command. For example:

```
symrdf - h
```

On UNIX systems, type `man <command>` to display the man page for the specified command. For example:

```
man symsrdf
```

Note: On UNIX systems, the SYMCLI man page directory (/usr/symcli/man/) must be specified in the MANPATH envronment variable. On Windows systems, the default directory for man pages is at C:\Program Files\EMC\symcli\man

Setting environmental variables

SYMCLI includes variables to streamline command line sessions.

Type the following command to display a list of variables that can be set for your SYMCLI session:

```
symcli -env
```

Type the following command to view the variables that are set:

```
symcli - def
```

Type the `setenv <VARIABLE_NAME> value` command to set a variable. For example:

```
setenv SYMCLI_VERBOSE 1
```

Type the `unsetenv <VARIABLE_NAME>` command to turn off a variable. For example:

```
unsetenv SYMCLI_VERBOSE
```

Presetting names and IDs

You can use the enviromental variables to preset the identity of Symmetrix objects so that you do not need to type them in the command line.

For example, the SYMCLI_ID variable sets the Symmetrix ID for all -sid arguments:

```
set env SYMCLI_SID 000192601365
```

Understanding SRDF pair states and links

IMPORTANT

Before you begin SRDF control operations, you must understand how SRDF devices and links work together to secure data within SRDF configurations.

An SRDF pair state encompasses:

- SRDF device state on the source (R1) device
- SRDF device state on the target (R2) device
- The number of invalid tracks on the R1 and R2 devices, and
- The SRDF link state between the R1 and R2 devices

Note: Invalid tracks are tracks that are not synchronized between members of a device pair. Invalid tracks are tracks that are *owed* to the SRDF device across the SRDF links. See "Invalid tracks in SRDF pairs" on page 30.

Figure 3 shows the various states SRDF devices and links can report.

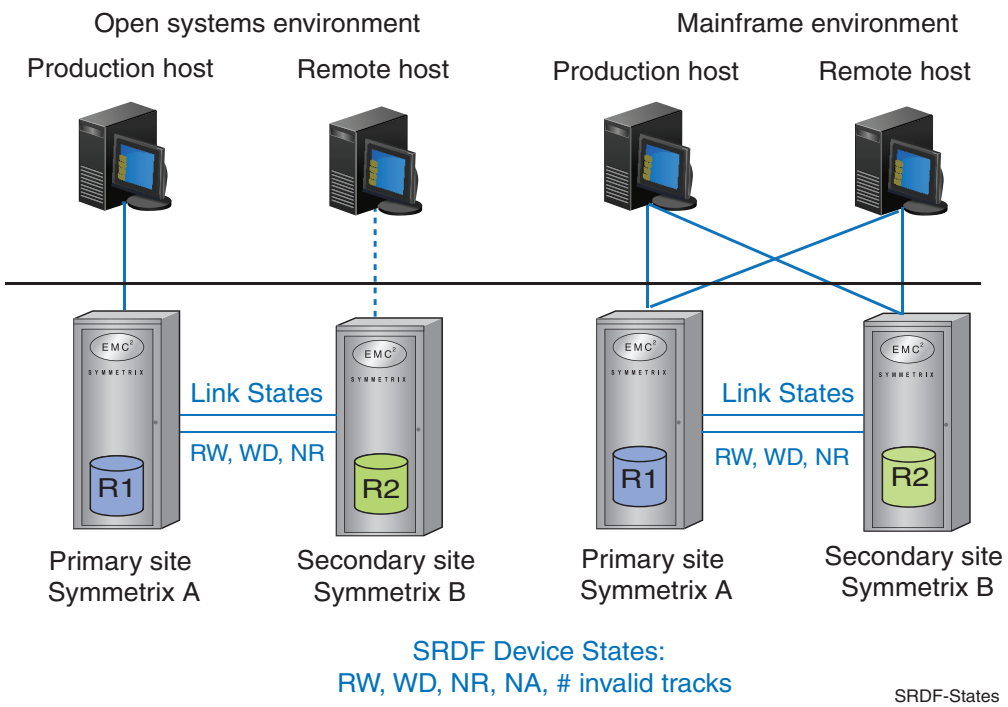


Figure 3 SRDF device and link states

The following legend provides details about the elements in Figure 3.

NR	Not Ready. Reads and writes are both disabled.
RW	Ready. Enabled for both reads and writes.
WD	Write Disabled. Enabled for reads but not writes.
NA	Not Available. Unable to report on correct state.

SRDF pair states

Before you can invoke an SRDF control operation, the SRDF pair state must be valid for that action.

See "SRDF Pair State Reference" on page 497 for a list of control actions and the prerequisite SRDF pair state for each action. Table 3 lists the state name and description of SRDF pair states.

See "Verifying SRDF modes and pair states" on page 37 for information on how to verify pair states.

Table 3 SRDF pair states (page 1 of 2)

State	Description
SyncInProg	A synchronization is currently 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 currently in a synchronized state: • The same content exists on the R2 as the R1. • There are no invalid tracks between the two pairs.
Split	The R1 and the R2 are currently ready to their hosts, but: • The links are not ready or, • Write disabled.
Failed Over	The R1 is currently not ready or write disabled. Operations have been failed over to the R2.
R1 Updated	The R1 is currently 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 currently not ready or write disabled to the host: • There are invalid local (R1) tracks on the source side, • Data is being copied from the R2 to the R1 device, and the links are ready.
Suspended	The SRDF links have been suspended and are not ready or write disabled. If the R1 is ready while the links are suspended, any I/O will accumulate as invalid tracks owed to the R2.
Partitioned	The SYMAPI is currently unable to communicate through the corresponding SRDF path to the remote Symmetrix. The Partitioned state may apply to devices within an RA group. For example, if SYMAPI is unable to communicate to a remote Symmetrix from an RA group, devices in that RA group will be marked as being in the Partitioned state. A half pair and a duplicate pair are also reported as Partitioned.
Mixed	A composite SYMAPI device group SRDF pair state. There are different SRDF pair states within a device group.

Table 3 SRDF pair states (page 2 of 2)

State	Description
Invalid	The default state when no other SRDF state applies. • 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	The R2 SRDF/A capable devices are in a consistent state. Consistent state signifies the normal state of operation for device pairs operating in asynchronous mode.
Transmit Idle	The SRDF/A session cannot push data in the transmit cycle across the link because the link is down.

Cascaded and concurrent pair states

In a cascaded relationship, control operations are only allowed for the pair R1->R21 when the R21->R2 pair is in a specific pair state, and vice versa.

In a concurrent relationship, there are two separate links, or legs, sending data from an R1 device to two separate R2 devices. You can perform a control operation on one of these legs only if the other leg is in a certain pair state.

"SRDF Pair State Reference" on page 497 provides information about control operations and their valid pair states for both cascaded and concurrent SRDF configurations.

Invalid tracks in SRDF pairs

Invalid tracks are tracks that are not synchronized between the two devices in an SRDF pair.

On both the source and target sides of an SRDF configuration, the Symmetrix array keeps an account of the tracks that are "owed" to the other side.

Remote invalids are tracks owed the remote member of the device pair.

For example, the logical connection between a R1 device to its R2 is suspended. If both devices are made write-accessible, hosts on both sides of the SRDF links can write to their respective devices, creating R2 invalids on the R1 side and R1 invalids on the R2 side. Each invalid track represents a track of data that has changed since the two sides were split. To re-establish the logical links between the R1 and R2, the invalid tracks must be resolved.

Resolution of invalid tracks depends on which control operation you perform. For example if you have remote invalids on both the R1 and R2 sides:

- An `establish` operation copies the modified R1 tracks to the R2 side. Any tracks that were modified on the R2 side are overwritten with data from corresponding tracks on the R1 side.
- A `restore` operation copies the modified R2 tracks to the R1 side. Any tracks that were modified on the R1 side are overwritten with data from corresponding tracks on the R2 side.

SRDF device and link state combinations

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

This depends on whether the SRDF state of the source (R1) side, the SRDF links, or the SRDF state of the target (R2) side has changed. Additionally, the state of a device can change if its front-end or back-end Integrated Directors change in the SRDF links.

Table 4 shows:

- 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[1]
R1 UpdInProg	Not Ready or WD	Ready (RW) or WD	Ready (RW)	>0[1]
Split	Ready (RW)	Not Ready or WD	Ready (RW)	—
SyncInProg	Ready (RW)	Ready (RW)	Not Ready or WD	>0
Suspended	Any status[2]	Not Ready or WD	Not Ready or Write Disabled	—
Partitioned[3]	Any status	Not Ready	Not Available	—
Partitioned[4]	Not Available	Not Ready	Any status	—
Mixed	[5]	[5]	[5]	—
Invalid[5]	Any status[6]	Any status	Any status	—
Consistent	Ready (RW)[6]	Ready (RW)	Not Ready or WD	0
Transmit Idle	Ready (RW)[6]	Ready (RW)	Not Ready or WD	—

1. Refers to invalid local (R1) tracks on source.
2. Any status value is possible (Ready, Not Ready, Write Disabled, or Not Available).
3. Viewed from the host locally connected to the source (R1) device
4. Viewed from the host locally connected to the target (R2) device.
5. When no other SRDF states apply, the state defaults to Invalid.
6. The combination of source SRDF, SRDF links, and target SRDF statuses do not match any other SRDF state; therefore, the SRDF state is considered Invalid.

Pinging Symmetrix arrays through SRDF links

You can ping Symmetrix arrays through SRDF links.

Type the `symrdf -rdf ping` command to determine if a Symmetrix array using SRDF links is up and running. For example, to ping Symmetrix array 123:

```
symrdf -rdf -sid 123 ping
```

The return codes tell you whether some or all of the Symmetrix arrays were successfully pinged. For more information on return codes, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Understanding and setting SRDF modes of operation

Each SRDF configuration operates in one of the following modes:

- synchronous
- semi-synchronous
- adaptive copy
- asynchronous.

SRDF modes of operation determine the following:

- How R1 devices are remotely mirrored to R2 devices across the SRDF links
- How I/Os are processed in an SRDF solution
- When Enginuity returns the acknowledgement to the production host that issued a write I/O command.

Setting the default SRDF mode

Prior to Solutions Enabler V7.4, the default SRDF mode is synchronous. If you create device pairs without setting a mode, the devices are in synchornous mode.

For Solutions Enabler V7.4 and higher, the default mode is adaptive copy disk. If you create device pairs without setting a mode, the devices are in adaptive copy disk mode.

Use the SYMAPI_DEFAULT_RDF_MODE parameter in the options file to modify the default mode.

Setting the mode for devices, groups, and device files

Use the `set` attribute with the `symrdf` command to set the SRDF mode on remotely-mirrored standard devices in a device group, composite group, or device file.

The syntax fo setting mode on a device group, composite group, and device file is as follows:

```
symrdf -g DgName set mode
symrdf -cg CgName set mode
symrdf -f[ile] FileName set mode -sid SymmID -rdfg GroupNumber
```

For more information, see “SRDF Control Operations” on page 47.

Use the `createpair` operation to assign the mode of operation to device pairs.

For more information, see “Creating dynamic SRDF device pairs” on page 100.

Synchronous

Synchronous mode maintains a real-time mirror image of data between the R1 and R2 devices. Data must be successfully stored in Symmetrix cache at both the primary and the secondary site before an acknowledgement is sent to the production host at the primary site.

Synchronous mode ensures that the source (R1) and target (R2) devices contain identical data.

To set the remotely mirrored pairs in the `prod` group to the synchronous mode, type:

```
symrdf -g prod set mode sync
```

Semi-synchronous

Note: Semi-synchronous mode is supported only in Engenuity versions earlier than 5773.

Semi-synchronous mode allows the R1 and R2 devices to be out of synchronization by one write I/O operation.

Data must be successfully written to the primary device before an acknowledgement is sent to the production host. Semi-synchronous mode does not allow another write I/O to the R1 device until the primary array receives an acknowledgement from the secondary array that the first write operation was received in Symmetrix cache at the secondary site:

- The Symmetrix array containing the source (R1) device informs the host of successful completion of the write operation when it receives the data.
- The SRDF (RA) director transfers each write to the target (R2) device as the SRDF links become available.
- The Symmetrix array containing the target (R2) device checks and acknowledges receipt of each write.
- If a new write is started for a source (R1) device before the previous write has completed to the target (R2) device, the Symmetrix array containing the source (R1) device temporarily disconnects from the I/O bus until the previous write operation is completed and acknowledged.
- When the previous write operation is completed and acknowledged, the remote Symmetrix array reconnects to the I/O bus and continues processing.

Note: In mainframe host environments, semi-synchronous mode is not supported if the Symmetrix array uses Fibre Connection (FICON) host connectivity. In mainframe Parallel Access Volumes/Multiple Allegiance (PAV/MA) environments, semi-synchronous mode is not recommended because it provides no performance benefit. If you want to use the increased parallelism offered by PAVs, run SRDF in synchronous mode.

To set all the remotely mirrored pairs in the device group `prod` to the semi-synchronous mode, type:

```
symrdf -g prod set mode semi
```

Adaptive copy

Adaptive copy modes allow the R1 and R2 devices to be more than one I/O out of synchronization. Unlike the asynchronous mode, adaptive copy modes do not guarantee a dependent-write consistent copy of data on R2 devices.

The amount of data (number of tracks) out of synchronization between the R1 and the R2 devices at any given time is determined by the maximum skew value. You can set the *maximum skew value* by using EMC host-based SRDF software.

As secondary modes of operation, adaptive copy modes revert to the specified primary mode of operation, which is either synchronous mode or semi-synchronous mode, when certain conditions are met.

There are two versions of adaptive copy mode:

- "Adaptive copy write pending"
- "Adaptive copy disk"

Adaptive copy write pending

When adaptive copy write pending (acp_wp) mode is enabled, the Symmetrix array acknowledges all writes to the source (R1) device as if it is a local device.

The amount of data (number of tracks) out of synchronization between the R1 and the R2 devices at any given time is determined by the maximum skew value. You can set the *maximum skew value* using EMC host-based SRDF software.

New data accumulates *in cache* until it is successfully written to the source (R1) device and the remote director has transferred the write to the target (R2) device.

This SRDF mode is designed to have little or no impact on performance between the host and the Symmetrix array containing the source (R1) device.

To turn on the adaptive copy write pending mode for the device group `prod`, type:

```
symrdf -g prod set mode acp_wp
```

To turn off adaptive copy write pending mode for the device group prod, type:

```
symrdf -g prod set mode acp_off
```

Adaptive copy disk

Adaptive copy disk (acp_disk) mode is designed to transfer large amounts of data without loss of performance.

Because the Symmetrix array cannot fully guard against data loss should a failure occur, EMC recommends:

- Use the adaptive copy disk mode to transfer the bulk of your data to target (R2) devices, and then
- Switch to synchronous mode to ensure full data protection.

When you set the SRDF mode to adaptive copy disk, the Symmetrix array acknowledges all writes to source (R1) devices as if they were local devices. New data accumulates on the source (R1) device and is marked by the source (R1) side as *invalid tracks* until it is subsequently transferred to the target (R2) device. The remote director transfers each write to the target (R2) device whenever link paths become available.

To turn on the adaptive copy disk mode for the `prod` group, type

```
symrdf -g prod set mode acp_disk
```

To turn the adaptive copy disk mode off for the `prod` group, type:

```
symrdf -g prod set mode acp_off
```

Adaptive copy disk skew

Skew is an attribute that defines the maximum number of invalid tracks supported by adaptive copy disk mode.

If the number of invalid tracks defined by the skew attribute is exceeded, the remotely-mirrored device switches to synchronous mode.

As soon as the number of invalid tracks drops below the skew threshold, the remotely-mirrored pair reverts back to adaptive copy mode.

Skew is configured at the device level and may be set to a value between 0 and 65,534 tracks. For devices with more than a 2 GB capacity drive, you can specify a value of 65,535 to indicate all tracks of any given drive.

To change the adaptive copy skew value to the number of tracks on device `BCV023` of group `prod`, type:

```
symrdf -g prod set acp_skew 65535 -bcv BCV023
```

The above command locks device `BCV023` into adaptive copy disk mode since the number of invalid tracks cannot exceed the maximum threshold of 65, 535.

To change the adaptive copy skew value to 30,000 tracks for device `BCV023` of group `prod`, type:

```
symrdf -g prod set acp_skew 30000 -bcv BCV023
```

Asynchronous

In asynchronous mode (SRDF/A), data is transferred from the source (R1) site in predefined timed cycles or *delta sets* to ensure that data at the remote (R2) site is *dependent write consistent*.

The Symmetrix array acknowledges all writes to the source (R1) devices as if they were local devices. Host writes accumulate on the source (R1) side until the cycle time is reached and are then transferred to the target (R2) device in one delta set. Write operations to the target device are confirmed when the current SRDF/A cycle commits the data to disk by successfully de-staging it to the R2 storage devices.

Because the writes are transferred in cycles, any duplicate tracks written to can be eliminated through Symmetrix *ordered write processing*, which transfers only the changed tracks within any single cycle.

The point-in-time copy of the data at the secondary site is slightly behind that on the primary site.

SRDF/A has little or no impact on performance at the primary site as long as the SRDF links contain sufficient bandwidth and the secondary system is capable of accepting the data as quickly as it is being sent across the SRDF links.

When you set the mode as asynchronous for an SRDF group, all devices in the group must operate in that mode. A subset of devices within the SRDF group cannot operate in synchronous mode.

To set the remotely mirrored pairs in the `prod` group to asynchronous mode, type:

```
symrdf -g prod set mode async
```

Note: A device status check is performed on all TimeFinder Snap and Clone device pairs in the group before the `set mode async` operation is allowed. Depending on the device pair state, asynchronous mode may not be allowed for devices employing either TimeFinder/Snap or TimeFinder/Clone operations. Appendix A explains the applicable device pair states for TimeFinder/Snap or TimeFinder/Clone operations.

"SRDF/Asynchronous operations" on page 116 provides additional information on operating in asynchronous mode.

Domino effect

Under normal operating conditions, the domino effect is disabled (not turned on).

When an SRDF device or link failure occurs, remotely mirrored devices continue processing I/Os from hosts.

New data written to the source (R1) or target (R2) device while its remote partner is unavailable or link paths are out of service are marked for later transfer.

When link paths are re-established or the device becomes available, resynchronization begins between the source (R1) and target (R2) devices.

To turn the domino effect off for the device group `prod`, type:

```
symrdf -g prod set domino off
```

Domino effect on

Domino effect mode ensures data on the source (R1) and target (R2) devices are always in sync. When the domino effect is turned on, and the Symmetrix array detects that one side of a remotely mirrored pair is unavailable, or that the SRDF link has failed, the array:

- Forces the source (R1) device to a not ready state, and
- Responds "*intervention required/unit not ready*" to the host

After the problem is corrected, the *not ready* device must be made *ready* to the host using the `symrdf ready` command.

If failed devices or links are still not available when the devices are made ready, the devices become not ready again.

The following restrictions apply to the domino effect:

- Domino effect mode cannot be enabled for SRDF/A-capable devices.
- When consistency protection is enabled on a device, the domino effect mode cannot be turned on, and vice versa.
- When domino effect is turned on for a device, you cannot use the `split` or `suspend` control operations because the devices are not ready.
- Regardless of whether domino effect is enabled, all SRDF links still fail when R1 devices become not ready.

To turn the device domino effect on for the `prod` device group, type:

```
symrdf -g prod set domino on
```

To make all the source (R1) side devices ready in the device group `prod`, type:

```
symrdf -g prod ready r1
```

Verifying SRDF modes and pair states

To query and verify SRDF operations performed on device groups, composite groups and device files, use the `symrdf query` and `symrdf verify` commands.

The examples in this section use a device group named “STAGING”, which contains device pairs operating in synchronous, adaptive copy disk, and adaptive copy write pending modes and are in the Synchronized, Suspended and Split states:

```
symrdf -g STAGING -rdfg 129 query

Device Group (DG) Name             : STAGING
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number              : 129 (80)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0080 RW       0        0 RW 0080 WD       0        0 S...  Synchronized
DEV002  0081 RW       0        0 NR 0081 WD       0        0 C.W.  Suspended
DEV003  0082 RW       0        0 NR 0082 RW       0        0 C.D.  Split
DEV004  0083 RW       0        0 RW 0083 WD       0        0 C.D.  Synchronized
DEV005  0084 RW       0        0 RW 0084 WD       0        0 C.D.  Synchronized
DEV006  0085 RW       0        0 RW 0085 WD       0        0 C.D.  Synchronized
DEV007  0086 RW       0        0 RW 0086 WD       0        0 C.D.  Synchronized
DEV008  0087 RW       0        0 RW 0087 WD       0        0 C.D.  Synchronized
DEV009  0088 RW       0        0 RW 0088 WD       0        0 C.D.  Synchronized
DEV010  0089 RW       0        0 RW 0089 WD       0        0 C.D.  Synchronized
DEV011  008A RW       0        0 RW 008A WD       0        0 C.D.  Synchronized
DEV012  008F RW       0        0 RW 008F WD       0        0 C.D.  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

Verifying SRDF modes

The `symrdf verify` command monitors and verifies the SRDF modes of pair states defined in device groups, composite groups, and device files.

Table 5 lists SRDF modes and the `symrdf verify` options:

Table 5 symrdf verify command options for verifying SRDF modes

SRDF mode	Command option	Abbreviated command option
Adaptive copy disk	-acp_disk	-acp_disk
Adaptive copy write pending	-acp_wp	-acp_wp
Asynchronous	-asynchronous	-async
Semi-synchronous	-semisynchronous	-semi
Synchronous	-synchronous	-sync

Note: Solutions Enabler supports abbreviated command options for SRDF modes. Solutions Enabler does not support abbreviated forms for SRDF pair states.

When verifying two or more SRDF modes at one time, Solutions Enabler logically ORs each mode to determine the result.

The following command verifies if any of the device pairs in the STAGING group are in asynchronous mode:

```
symrdf -g STAGING -rdfg 129 verify -async
```

Since none of the device pairs in the STAGING group are in asynchronous mode, the following message displays:

```
None of the devices in the group 'STAGING' are in 'Asynchronous'
mode.
```

Use the following command to check if the device pairs in the STAGING group are in asynchronous OR synchronous mode:

```
symrdf -g STAGING -rdfg 129 verify -async -sync
```

Since some of the device pairs in the STAGING group are using synchronous mode but none are using asynchronous mode, the following message displays:

```
Not All devices in the group 'STAGING' are in 'Asynchronous,
Synchronous' modes.
```

Use the following command to verify if the device pairs in the STAGING group are in asynchronous, synchronous mode, OR adaptive copy disk mode:

```
symrdf -g STAGING -rdfg 129 verify -async -sync -acp_disk
```

Since some of the device pairs in the STAGING group are using synchronous and adaptive copy disk modes but are not using asynchronous mode, the following message displays:

```
Not All devices in the group 'STAGING' are in 'Asynchronous,
Synchronous, Adaptive Copy Disk' modes.
```

Use the following command to verify if the device pairs in the STAGING group are in asynchronous, synchronous, adaptive copy disk, OR adaptive copy write pending mode:

```
symrdf -g STAGING -rdfg 129 verify -async -sync -acp_disk -acp_wp
```

Even though none of the device pairs are using asynchronous mode, Solutions Enabler displays the following message because all device pairs in the STAGING group are using synchronous, adaptive copy disk, OR adaptive copy write pending mode:

```
All devices in the group 'STAGING' are in 'Asynchronous, Synchronous,
Adaptive Copy Disk, Adaptive Copy Write Pending' modes.
```

Verifying SRDF pair states

When verifying two or more SRDF pair states at one time, Solutions Enabler logically ORs each pair state to determine the result.

Use the following command to verify if any of the device pairs in the STAGING group are in the Consistent state:

```
symrdf -g STAGING -rdfg 129 verify -consistent
```

Since none of the device pairs in the STAGING group are in the Consistent state, the following message displays:

```
None of the devices in the group 'STAGING' are in 'Consistent' state.
```

Use the following command to check if the device pairs in the STAGING group are in the Consistent OR Split state:

```
symrdf -g STAGING -rdfg 129 verify -consistent -split
```

Since some of the device pairs are in the Split state but are not in the Consistent state, the following message displays:

```
Not All devices in the group 'STAGING' are in 'Consistent, Split'
states.
```

Use the following command to verify if the device pairs in the STAGING group are in the Consistent, Split, OR Suspended states:

```
symrdf -g STAGING -rdfg 129 verify -consistent -split -suspended
```

Since some of the device pairs in the STAGING group are in the Split and Suspended states but are not in the Consistent state, the following message displays:

```
Not All devices in the group 'STAGING' are in 'Consistent, Split,
Suspended' states.
```

Use the following command to verify if the device pairs in the STAGING group are in the Consistent, Split, Suspended, OR Synchronized states:

```
symrdf -g STAGING -rdfg 129 verify -consistent -split -suspended
-synchronized
```

Even though none of the device pairs are in the Consistent state, Solutions Enabler displays the following message because all device pairs in the STAGING group are in the Split, Suspended, OR Synchronized state:

```
All devices in the group 'STAGING' are in 'Consistent, Split,
Suspended, Synchronized' states.
```

Verifying SRDF modes and pair states simultaneously

When verifying both SRDF states and modes at the same time, Solutions Enabler logically ORs the states, logically ORs the modes, and then logically ANDs these two results.

Use the following command to verify that the device pairs in the STAGING group are using synchronous, adaptive copy disk, OR adaptive copy write pending mode AND are in the Synchronized, Suspended, OR Split state:

```
symrdf -g STAGING -rdfg 129 verify -sync -acp_disk -acp_wp
-synchronized -suspended -split
```

Since all device pairs in the STAGING group are using synchronous, adaptive copy disk, OR adaptive copy write pending mode AND are in the Synchronized, Suspended, OR Split state, the following message displays:

```
All devices in the group 'STAGING' are in 'Synchronized, Suspended,
Split' states and 'Synchronous, Adaptive Copy Disk, Adaptive Copy
Write Pending' modes.
```

Use the following command to verify that the device pairs in the STAGING group are using adaptive copy disk OR adaptive copy write pending mode AND are in the Synchronized, Suspended, OR Split state:

```
symrdf -g STAGING -rdfg 129 verify -acp_disk -acp_wp -synchronized
-suspended -split
```

Since some device pairs in the STAGING group are using synchronous mode, the following message displays:

```
Not All devices in the group 'STAGING' are in 'Synchronized,
Suspended, Split' states and 'Adaptive Copy Disk, Adaptive Copy Write
Pending' modes.
```

Use the following command to verify that the device pairs in the STAGING group are using synchronous, adaptive copy disk, OR adaptive copy write pending mode AND are in the Consistent state:

```
symrdf -g STAGING -rdfg 129 verify -sync -acp_disk -acp_wp
-consistent
```

Since none of the device pairs in the STAGING group are in the Consistent state, the following message displays:

```
None of the devices in the group 'STAGING' are in 'Consistent' state
and 'Synchronous, Adaptive Copy Disk, Adaptive Copy Write Pending'
modes.
```

Note: You can also use the `symrdf verify -enabled` command to validate whether device pairs are enabled for consistency protection. You can also use `-enabled` with any pair state and SRDF mode combination in the `symrdf verify` command, as explained in "Checking if device pairs are enabled for consistency protection" on page 210.

Operational considerations

This section provides information to be aware of before invoking the SYMCLI SRDF control operations.

Symmetrix array access rights

A host must have specific access rights to a Symmetrix array to perform certain control operations. Table 6 shows the control operation and its required Symmetrix array access rights. Unless otherwise noted, the access type of SRDF is required for all SRDF control operations.

Table 6 Access rights required by a Symmetrix array

Operations	Required access rights
`symrdf set -rdf`	CFGSYM or SRDF
`symrdf set -rdfa`	CFGSYM or SRDF
`symrdf set rdfa_dse`	CFGSYM or SRDF
`symrdf set rdfa_pace`	CFGSYM or SRDF
`symrdf addgrp`	CFGSYM
`symrdf modifygrp`	CFGSYM
`symrdf removegrp`	CFGSYM
`symqos set IO`	CFGSYM
`symqos reset IO`	CFGSYM

SRDF operations and copy sessions

Certain SRDF operations are not allowed within Symmetrix arrays employing either TimeFinder/Snap or TimeFinder/Clone operations, which use copy session pairs. The availability of some SRDF actions depends on the current pair state of the TimeFinder/Snap or TimeFinder/Clone copy session devices.

Refer to Appendix A for a description of the TimeFinder/Snap and TimeFinder/Clone pair states, and which SRDF operations are available within each state.

Migrating data from R1 to a larger R2 device

You can copy data from an R1 device to a larger R2 device but the following restrictions apply:

- All swap and SRDF/Star operations are blocked.
- If `SYMAPI_RDF_CREATEPAIR_LARGER_R2` is set to `DISABLE` in the options file, all `createpair` operations are blocked.
- Data mirrored to a larger R2 device cannot be restored back to its R1 device.
- Concatenated metadevices are not supported but striped metadevices are supported.

Note: Depending on the type of file system and attached host, certain host-dependent operations may be required to access data migrated to a larger R2 device.

Preventing synchronization actions

For some sites, it may be desirable to block users on a host from performing either an establish or restore operation on any of the Symmetrix devices. The sync direction parameter (`SYMAPI_SYNC_DIRECTION`) in the options file allows you to confine SRDF and TimeFinder operations to only establish or restore actions.

You can block a user on a host from executing a restore or an establish action using the following form:

```
SYMAPI_SYNC_DIRECTION=ESTABLISH | RESTORE | BOTH
```

`ESTABLISH` confines the possible operations to just establish actions.

`RESTORE` confines the possible operations to just restore actions, which includes (allows) restore, failback, R1 update actions.

`BOTH` is the default, which does not restrict any SRDF or TimeFinder actions.

Device external locks

SYMAPI/SYMCLI uses *device external locks* in the Symmetrix array to lock BCV pairs during TimeFinder control operations and to lock SRDF device pairs during SRDF control operations.

To list a range of Symmetrix devices (`0000` to `000A`) that have a device external lock, type:

```
symdev list -sid 870 -devs 0000:000A -lock
```

Enabling SRDF software and hardware compression

Compression minimizes the amount of data to be transmitted over an SRDF link. Solutions Enabler provides a way to enable, disable, and query software and hardware compression on an SRDF group. Compression is available on:

- Enginuity 5875 or higher - Hardware compression on GigE directors and software compression on Fibre and GigE directors using Solutions Enabler.
- Enginuity 5876 or higher - Hardware compression on Fibre directors using Solutions Enabler.

You can enable software and hardware compression on the R1 and R2 sides but the actual compression happens from the side initiating the I/O, which is typically the R1 side. Since most I/O requests are initiated from the R1 side, make sure you enable compression on this side.

Use the following syntax to set hardware and software compression for an SRDF group:

```
symrdf -sid <SymmID> -rdfg <GrpNum> [-v] [-symforce]
   [-noprompt] [-i <Interval>] [-c <Count>]

.............

set rdfg
    [-limbo <LinkLimboValue>]
    [-domino <State>]
    [-autolink_recovery <State>]
    [-hwcomp <State>]
    [-swcomp <State>]
    [-both_sides]
```

For more information on `set rdfg`, see "Setting SRDF group attributes" on page 98.

For example, to turn on software compression and turn off hardware compression on both sides of SRDF group 12, type:

```
symrdf -sid 134 -rdfg 12 set rdfg -swcomp on -hwcomp off -both_sides
```

To determine if SRDF software and hardware compression is enabled for an SRDF group, issue the `symcfg list -rdfg` command.

For example, to view if software or hardware compression is enabled for SRDF group 12 on Symmetrix 432, type:

```
symcfg list -sid 432 -rdfg 12
```

SRDF/A and the consistency exempt option

When an SRDF group is supporting an active SRDF/A session, control operations must be targeted at all device pairs in the session, or the session must first be made inactive by suspending the links between all device pairs in the session before attempting to control a subset of the device pairs.

The consistency exempt feature, available with EMC Enginuity® 5773.150 and higher, relaxes that restriction. Devices marked consistency exempt can, in some cases, be controlled independently of other devices in the active SRDF/A session. The `-cons_exempt` option flags devices targeted by the command as consistency exempt. Enginuity automatically clears the consistency exempt status when the affected device pairs have become consistent and two cycle switches have subsequently occurred.

Mixed-mode workloads on an SRDF director

For Symmetrix arrays running Enginuity 5876 and higher, Solutions Enabler supports the following mixed-mode workloads on the same SRDF director (RA):

- Synchronous I/Os
- Asynchronous I/Os
- Copy I/Os

To set the percentage of the SRDF director CPU resources assigned to synchronous, asynchronous, and copy I/Os, use the following `symqos` command syntax:

```
symqos -RA -sid <SymmID>

   enable -io
```

```
      disable -io

   symqos -RA -sid <SymmID>

      set IO -default
      -sync <SyncPercent> -async <AsyncPercent> -copy <CopyPercent>

      set IO -dir <# | ALL>
      -sync <SyncPercent> -async <AsyncPercent> -copy <CopyPercent>

      reset IO -dir <# | ALL>

   symqos -RA [-sid <SymmID>]

      list -io
```

Examples

To enable the workload percentage settings for synchronous, asynchronous, and copy I/Os on Symmetrix 1234, type:

```
symqos -RA -sid 1234 enable -io
```

To set the default settings of the workload percentages for all directors on Symmetrix 1234 to 60% for Synchronous I/Os, 30% for asynchronous I/Os and 10% for copy I/Os, type:

```
symqos -RA -sid 1234 set IO -default -sync 60 -async 30 -copy 10
```

To set the settings of the workload percentages on director 8G of Symmetrix 1234 to 50% for synchronous I/Os, 30% for asynchronous I/Os, and 20% for copy I/Os, type:

```
symqos -RA -sid 1234 -dir 8G set IO -sync 50 -async 30 -copy 20
```

To reset the customized settings of the workload percentages back to the default settings on director 8G of Symmetrix 1234, type:

```
symqos -RA -sid 1234 -dir 8G reset IO
```

Workload percentages must add up to a total of 100%. When you set the workload percentages for a director, these values are used until you reset them for that director, and then the Symmetrix-level distributions are used.

For information on the `symqos` command syntax, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*.

FAST VP SRDF coordination

SRDF coordination instructs FAST VP to factor the R1 device statistics into the move decisions that are made on the R2 device.

You can set this attribute on a storage group, even when there are no SRDF devices in the storage group. To set the SRDF coordination attribute, use the `symfast` command, as follows:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
             -fp_name FastPolicyName
   associate -sg SgName -priority PriorityValue
            [-rdf_coordination <ENABLE | DISABLE>]
```

For FAST VP SRDF coordination, both the R1 and the R2 devices must reside on Symmetrix arrays running Enginuity 5876 or higher.

For information on FAST and FAST VP, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*.

CHAPTER 2
SRDF Control Operations

This chapter describes control operations to establish, manage, and list components comprising an SRDF configuration. Most of these operations are performed using the SYMCLI `symrdf` command. Topics include:

- Device groups, device files and composite groups 48
- Performing SRDF control operations 50
- Displaying SRDF information 86

Device groups, device files and composite groups

Most SRDF operations are performed on devices in device groups (-g), device files (-file), and composite groups (-cg).

The *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide* explains how to create device groups, device files, and composite groups.

Device groups

An SRDF device group is a user-defined device group comprised of SRDF devices from a single Symmetrix array. At the time of creation, a device group must be defined as type REGULAR, RDF1, RDF2, RDF21, or ANY, and may contain various device lists for standard, BCV, virtual (VDEV), and remote devices. If the group type is defined as RDF1, RDF2, or RDF21, the group is considered an SRDF device group.

Note: A device group of any SRDF type can change its type because of a `symrdf` control operation. For example, an RDF1 DG can change to an RDF2 when the device personalities are swapped. SRDF control operations cannot change the type of an ANY device group but can affect the devices in that device group (such as the `suspend`, `establish`, and `swap` operations).

By default, a device cannot belong to more than one device group. However, you can change this default behavior to allow a device to belong to multiple groups by enabling the SYMAPI_ALLOW_DEV_IN_MULT_GRPS parameter in the Symmetrix `options` file. You can use device groups to identify and work with a subset of available Symmetrix devices, obtain configuration, status, and performance statistics on a collection of related devices, or issue control operations that apply to all devices in the specified device group.

A device group can be a member of one or more composite groups.

Device files

The device file option directs the specified operation in the `symrdf` command to a device file. The device file contains device pairs (*SymDevnames*) listing a pair on each line. Device files can include comment lines that begin with the pound sign (#). The following example illustrates the file format, which specifies three device pairs:

```
 00A1 0103
 00A2 0104
#00A3 0105 (To be reinstalled later)
 00B1 0106
```

When using this option, specify an SRDF group to which all devices in the first column can belong. Also include a target Symmetrix ID or set the environmental variable `SYMCLI_SID`. These options allow you to operate on Symmetrix arrays and remote BCV pairs beyond the first SRDF multi-hop.

Composite groups

A composite group is a user-defined group of devices or device groups that can span multiple Symmetrix arrays and SRDF groups. The composite group type may be defined as REGULAR, RDF1, RDF2, RDF21, or ANY, and may contain various device lists for standard, BCV, virtual (VDEV), RBCV, BRBCV, second hop standard, and second hop BCV.

Note: A composite group of any SRDF type can change its type because of a `symrdf` control operation. For example, an RDF1 CG can change to an RDF2 when the device personalities are swapped. SRDF control operations (such as the `suspend`, `establish`, and `swap` operations) cannot change the type of an ANY composite group but can affect the devices in that CG.

You can enable a composite group for remote database consistency, as explained in "SRDF Consistency Group Operations" on page 199.

Querying and verifying

To query and verify SRDF operations performed on device groups, device files, and composite groups, use the `symrdf query` and `symrdf verify` commands. See "Querying and Verifying with SRDF Commands" on page 349 for command examples.

Performing SRDF control operations

When SRDF control operations are initiated, the SRDF state of the device pair is first verified. If the device pair is not in a valid SRDF state to initiate the control operation, the action is blocked. For a list of control actions and the required SRDF pair states, refer to "SRDF Pair State Reference" on page 497.

Table 7 lists the SRDF control operations explained in this chapter. For information on dynamic SRDF control operations, such as `symrdf createpair` and `symrdf addgrp`, see "Dynamic SRDF Operations" on page 91.

Table 7 SRDF control operations (page 1 of 3)

Control operation	symrdf argument	Result
Activate or deactivate SRDF/A Delta Set Extension (DSE) page 56	activate -rdfa_dse deactivate -rdfa_dse	Activates or deactivates SRDF/A DSE.
Activate or deactivate SRDF/A write pacing page 57	activate -rdfa_wpace deactivate -rdfa_wpace activate -rdfa_devpace deactivate -rdfa_devpace activate -rdfa_pace deactivate -rdfa_pace	Activates or deactivates SRDF/A group-level write pacing. Activates or deactivates SRDF/A device-level write pacing. Activates or deactivates SRDF/A group-level and SRDF/A device-level write pacing at the same time.
Cleanup incomplete SRDF/A data page 59	msc_cleanup	Initiates a cleanup operation to discard any incomplete SRDF/A data to maintain dependent write consistency.
Enable or disable consistency protection page 59	enable disable	Enables or disables consistency protection for SRDF/A capable devices.
Establish an SRDF pair (full) page 59	establish -full	Establishes remote mirroring and initiates a full data copy from the source (R1) device to the target (R2) device. Use this for: • Initial synchronization of SRDF mirrors. • Replacement of a failed drive on the R2 side.
Establish an SRDF pair (incremental) page 61	establish	Establishes remote mirroring and initiates an incremental data copy from the source (R1) device to the target (R2) device. Use this for resynchronization of SRDF mirrors after a split and if you can discard the target data.
Failback page 63	failback	Switches data processing from the target side (R2) back to the source (R1) side. Use this to return the source site from the target site after resolving the cause of a failure.
Failover page 66	failover	Switches data processing from the source (R1) side to the target (R2) side. Use this when a failure occurs on the source side.

Table 7 SRDF control operations (page 2 of 3)

Control operation	symrdf argument	Result
Invalidate R1 mirror page 67	invalidate r1	Invalidates all tracks on the source (R1) side so that they can be copied over from the target (R2) side.
Invalidate R2 mirror page 68	invalidate r2	Invalidates all tracks on the target (R2) side so that they can be copied over from the source (R1) side.
Make ready the R1 mirror page 69	ready r1	Sets the source (R1) device to be SRDF ready to its local host.
Make ready the R2 mirror page 69	ready r2	Sets the target (R2) device to be SRDF ready to its local host.
Make the R1 mirror not ready page 69	not_ready r1	Sets the source (R1) device to be SRDF not ready to its local host.
Make the R2 mirror not ready page 70	not_ready r2	Sets the target (R2) device to be SRDF not ready to its local host.
Merge the track tables of the R1 and R2 devices page 70	merge	Merges the track tables between the source (R1) and the target (R2) side.
Move one-half of an SRDF pair page 71	half_movepair	Moves one-half of the SRDF device pair to a different SRDF group. **Note:** If the RA ends up supporting more than 64K devices in the new SRDF group, this operation fails.
Move SRDF device pairs page 71	movepair	Moves the SRDF device pair to a different SRDF group. **Note:** If the RA ends up supporting more than 64K devices in the new SRDF group, this operation fails.
Read/write disable target device page 71	rw_disable r2	Read/write disables the target (R2) device to its local host.
Refresh R1 mirror page 72	refresh r1	Marks any changed tracks on the source (R1) side to be refreshed from the R2 side.
Refresh R2 mirror page 73	refresh r2	Marks any changed tracks on the target (R2) side to be refreshed from the R1 side.
Restore from a target device page 73	restore -full	Resumes remote mirroring and initiates a full data copy from the target (R2) device to the source (R1) device. Use this for: • Initial (reverse) synchronization of SRDF mirrors. • Replacement of a failed drive on the R1 side.
Restore from a target device (incremental) page 75	restore	Resumes remote mirroring and initiates an incremental data copy from the target (R2) device to the source (R1) device. Use this for resynchronizing SRDF mirrors after a split and if you can discard the source data.
Resume SRDF links page 76	resume	Resumes I/O traffic on the SRDF links for the remotely mirrored SRDF pairs in the group.

Table 7 SRDF control operations (page 3 of 3)

Control operation	symrdf argument	Result
Split an SRDF pair page 77	split	Stops remote mirroring between the source (R1) device and the target (R2) device. The target device is made available for local host operations. Use this when both sides require independent access, such as for testing purposes.
Suspend SRDF links page 79	suspend	Suspends I/O traffic on the SRDF links for the remotely mirrored SRDF pairs in the group.
Swap SRDF pairs page 80	swap	Swaps the SRDF personality of the designated dynamic SRDF pair. Source R1 devices become target R2 devices and target R2 devices become source R1 devices.
Swap one-half of an SRDF pair page 80	half_swap	Swaps the SRDF personality of one half of the designated dynamic SRDF pair. Source R1 devices become target R2 devices or target R2 devices become source R1 devices.
Update R1 mirror page 81	update	Updates the source (R1) side with the changes from the target (R2) side while the target (R2) side is still operational to its local hosts. Use this to synchronize the R1 side with the R2 side as much as possible before performing a failback, while the R2 side is still online to the host.
Write disable source device page 83	write_disable r1	Write disables the source (R1) device to its local host.
Write disable target device page 84	write_disable r2	Write disables the target (R2) device to its local host.
Write enable source device page 84	rw_enable r1	Write enables the source (R1) device to its local host.
Write enable target device page 84	rw_enable r2	Write enables the target (R2) device to its local host.

Command option descriptions

Table 8 describes the basic options used in the SRDF control operations. Refer to the `symrdf` man page for descriptions of all the options.

Table 8 Basic options of the symrdf command (page 1 of 3)

Command option	Definition
-all	Targets the SRDF action at all devices in the device group, which includes standard SRDF devices and any BCV SRDF devices that are locally associated with the device. When used with `list`, the `-all` option shows all SRDF mirrors of the selected devices. **Note:** The -all flag is not supported for SRDF control operations on device groups or composite groups with type ANY.
-bcv	Targets the specified BCV devices associated with a device or composite group and are configured as SRDF BCV devices. By default, only the SRDF standard devices are affected by the SRDF control operations.
-brbcv	Targets the SRDF action at the specified remotely associated SRDF (Hop 2) BCV devices that can be paired with the remote mirrors of the local BCV devices.
-both_sides	Targets the SRDF control operation at both sides of an SRDF link.
-bypass	Causes the SRDF control operation to bypass existing Symmetrix exclusive locks. Use this option ONLY if no other SRDF operation is in progress at either the local and/or remote Symmetrix arrays.
-c	Counts the number of times to display or to attempt acquiring exclusive locks on the Symmetrix host database, the local Symmetrix array, and the remote Symmetrix arrays. If the (`-c`) option is not specified and an interval (`-i`) is specified, the program loops continuously to produce infinite redisplays, or until the SRDF control or `set` operation starts.
-cg	Specifies the composite group for SRDF operations.
-cons_exempt	For an SRDF group supporting an active SRDF/A session, allows devices to be added, removed, or suspended without affecting the state of the SRDF/A session or requiring that other devices in the session be suspended to perform the control operation. When used with `list`, this option shows devices that are consistency exempt or that are paired with devices that are consistency exempt.
-fibre	Uses the Fibre Channel communication protocol.
-file	Specifies the device file for SRDF operations.
-force	Performs the control operations on SRDF devices that are not in the expected state for a control operation. By using this option, the control operation is attempted, regardless of the pair state of the SRDF devices, and according to the rules in Table 55 on page 498.
-format	Used with `createpair` to clear all tracks on the R1 and R2 sides, ensuring no data exists on either side, and makes the R1 read write to the host.
-full	Requests a full establish or restore operation.
-g	Specifies the device group for SRDF operations.
-h	Provides brief, online help.
-hop2	Targets the group's second-hop (R21->R2) devices in a cascaded SRDF relationship. "Hop 2 controls" on page 156 provides more in formation.
-hwcomp	Enables or disables hardware compression, which minimizes the amount of data to transmit over an SRDF link.
-i	Executes a command at repeat intervals to display information or to attempt to acquire an exclusive lock on the Symmetrix host database, the local Symmetrix, and the remote Symmetrix arrays. The default interval is 10 seconds. The minimum interval is 5 seconds.
-immediate	Applies only to SRDF/A-backed devices. Causes `failover`, `split`, and `suspend` actions to drop the SRDF/A session immediately.
-label	Specifies a label for a dynamic SRDF group.

Table 8 Basic options of the symrdf command (page 2 of 3)

Command option	Definition
-noecho	Suppresses the display of progress status information.
-noprompt	Suppresses the message asking you to confirm an SRDF control operation.
-nowd	Bypasses the check to ensure the target of the operation is not writable by the host.
-offline	Obtains the data strictly from the configuration database. No connections are made to any Symmetrix arrays. The `symrdf` command uses information previously gathered from the Symmetrix array and held in the Symmetrix host database as opposed to interrogating the Symmetrix array directly. The offline option can alternatively be set by assigning the environment variable SYMCLI_OFFLINE to 1.
-rdfa_devpace	Indicates the operation affects the SRDF/A device-level write pacing feature.
-rdfa_dse	Indicates the operation affects the SRDF/A Delta Set Extension (DSE) feature.
-rdfa_pace	Indicates the operation affects both the group-level and the device-level components of the SRDF/A write pacing feature.
-rdfa_wpace	Indicates the operation affects the SRDF/A group-level write pacing feature.
-rdfa_wpace_exempt	Excludes the specified devices from SRDF/A group-level write pacing.
-rdfg	Targets a specific SRDF group number.
-rdf_mode	Used in `createpair` to set the SRDF mode of device pairs to one of the following: synchronous (sync), semi-synchronous (semi), asynchronous (async), adaptive copy disk mode (acp_disk), or adaptive copy write Pending mode (acp_wp).
-refresh	Marks the source (R1) devices or the target (R2) devices to refresh from the remote mirror.
-remote	Requests a remote data copy with the `failback`, `restore`, `resume`, `createpair` and `update` actions. When the concurrent links are ready, data is also copied to the concurrent SRDF mirror. For these actions to execute, use this option or suspend the concurrent links.
-remote_rdfg	Specifies the SRDF group number for the remote Symmetrix array.
-remote_sid	Specifies the remote Symmetrix array ID.
-restore	Used with `failover` to swap the R1 and R2 and restore the invalid tracks on the new R2 side (formerly R1) to the new R1 side (formerly R2). For more information, refer to "Issuing dynamic failover restore" on page 112.
-rp	Used with `-establish|-restore, createpair, failback, merge, restore, resume, update`, and `refresh`, to allow the operation even when one or more devices are tagged for RecoverPoint. When used with refresh, only allowed for refresh R1.
-rrbcv	Targets the SRDF action at the specified remotely associated SRDF (Hop 2) BCV devices, which can be paired with the remote mirrors of the local standard devices.
-sid	Specifies the local Symmetrix array ID.
-swcomp	Enables or disables software compression, which minimizes the amount of data to transmit over an SRDF link.

Table 8 Basic options of the symrdf command (page 3 of 3)

Command option	Definition
-symforce	Requests that the Symmetrix array force an operation by overriding all instances causing the array to reject an operation. The SYMAPI_ALLOW_RDF_SYMFORCE setting in the options file must be set to TRUE to use -symforce. With -symforce, a split command executes on an SRDF pair, even during a sync in progress state. **Note:** Use caution when applying this option as data can become lost or corrupted.
-until	Checks the number of invalid tracks that are allowed to build up from the active R2 local I/O before another update (R2 to R1) copy is retriggered. The update sequence loops until the invalid track count is less than the number specified for the `-until` value. Refer to "Continuous R1 updates" on page 82 for more information.
-v	Provides more detailed, verbose command output.

Activate and deactivate SRDF/A Delta Set Extension

Beginning with Enginuity 5772, you can activate and deactivate the SRDF/A Delta Set Extension (DSE) feature.

SRDF/A DSE extends the cache space available for an SRDF/A session by off-loading some or all of its cycle data from cache to preconfigured disk storage or pools, which are similar to SAVE device pools.

Note: The diskless cascaded SRDF feature is supported in conjunction with the SRDF/A DSE feature (see "SRDF/Extended Distance Protection" on page 161).

For detailed information on SRDF/A DSE, see "Managing SRDF/A Delta Set Extension pools" on page 128.

You can activate SRDF/A DSE in the following ways:

- When the SRDF/A group parameter `rdfa_dse_autostart` is set to ENABLE, SRDF/A becomes active when the SRDF/A session is activated.
- When the SRDF link status is Read Write, you can change SRDF/A DSE status for a device group, composite group, or file using the following commands:

  ```
  symrdf [-g DgName | -cg CgName | -f FileName]
  activate | deactivate -rdfa_dse
  ```

  This activates or deactivates SRDF/A DSE for groups on both the R1 and R2 sides.

  The SRDF links must be in asynchronous mode and SRDF/A must be active for activate or deactivate actions to succeed.
- When the SRDF link status is Read Write, you can change the SRDF/A DSE status using RA group operations as follows:

```
symrdf -sid SymmID -rdfg GrpNum [-v][-noprompt] [-i Interval] [-c Count]

                activate   -rdfa_dse  [-both_sides]
                deactivate  -rdfa_dse [-both_sides]
```

  When the `-both_sides` option is included, SRDF/A DSE is activated or deactivated for groups on both the source and target sides. Otherwise, the activate/deactivate is only performed on the source side.
- When SRDF/A DSE is active for an SRDF group, setting the group mode to `sync` or `acp` does not require the extra step of deactivating SRDF/A DSE. Deactivating SRDF/A in a group automatically deactivates SRDF/A DSE for that group.

Activate SRDF/A DSE with Dynamic Cache Partitioning

SRDF/A Delta Set Extension can be activated when dynamic cache partitioning is enabled, as follows:

```
symrdf <type> activate -rdfa_dse
```

The *type* can be `-dg`, `-cg`, `-file`, or `-rdfg`.

The requirements for activating SRDF/A DSE with dynamic cache partitioning are:

- All devices in the SRDF/A session must be in the same DCP.

- The `rdfa_dse_threshold` must be set, and must be lower than the `rdfa_cache_percentage` setting.
- The SRDF group must have at least one associated DSE pool with SAVE devices enabled.

The R1 and R2 cache usage is now reported as a percent of DCP Write Pending Limit.

Activate and deactivate SRDF/A write pacing

SRDF/A write pacing extends the availability of SRDF/A by preventing conditions that result in cache overflow on both the R1 and R2 sides. Solutions Enabler provides two types of write pacing: group-level write pacing and device-level write pacing, which can be active at the same time during an SRDF/A session. Because this feature is dynamic, it is different from similar features such as SRDF/A DSE.

You can activate and control group-level and device-level write pacing individually or simultaneously at the group, device group, composite group, or file level on the R1 side. Each has an autostart capability that automatically activates write pacing whenever an SRDF/A session becomes active. If an SRDF group has both group-level and device-level pacing configured to autostart, both are activated when the SRDF/A session becomes active.

For details on configuring group-level and device-level write pacing, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*. For information on using group-level and device-level write pacing, see "Using SRDF/A write pacing" on page 133.

The following is syntax for activating and deactivating SRDF/A write pacing at the device-group level:

```
symrdf -g DgName [-v | -noecho] [-force] [-symforce]

   activate <-rdfa_dse | -rdfa_pace | -rdfa_wpace | -rdfa_devpace> |
   -rdfa_wpace_exempt [<LdevName> [<LevdevName>....]]

   deactivate <-rdfa_dse | -rdfa_pace | -rdfa_wpace | -rdfa_devpace>|
   -rdfa_wpace_exempt [<LdevName> [<LevdevName>....]]
```

General requirements

The following are general requirements for SRDF/A write pacing:

- The group-level pacing function is supported on Symmetrix arrays running on Enginuity 5874.207.166 and higher.
- The device-level pacing function is supported on Symmetrix arrays running on Enginuity 5875 and higher.
- The `activate` argument requires that the SRDF/A session be active and contain at least one participating device. These requirements do not apply to the autostart capability.

Controlling group-level and device-level write pacing individually

This section provides examples on controlling group-level and device-level write pacing individually.

For example, to activate group-level write pacing for SRDF group 76, type:

```
symrdf -sid 123 -rdfg 76 activate -rdfa_wpace
```

To exempt `DEV001` in the `prod` group from SRDF/A write pacing, type:

```
symrdf -g prod -rdfg 76 -rdfa_wpace_exempt DEV001
```

To deactivate device-level write pacing for `DEV012` in the `prod` device group, type:

```
symrdf -g prod deactivate -rdfa_devpace DEV012
```

Controlling group-level and device-level write pacing simultaneously

After activating group-level and device-level write pacing at the same time, Enginuity monitors both the SRDF link performance of the SRDF/A session and the performance of the devices on the R2 side.

When controlling group-level and device-level write pacing simultaneously, the following are required:

- The `symrdf activate/deactivate -rdfa_pace` command is targeted at all devices in the SRDF group.
- The R1 Symmetrix array is accessible.
- The SRDF/A session being controlled is active and contains at least one participating device.
- The `symrdf deactivate -rdfa_pace` command requires the following:
  - The R2 Symmetrix array is accessible to verify that there are no TimeFinder/Snap or TimeFinder/Clone sessions off the R2 devices before deactivating device-level pacing.
  - If the SRDF/A session is in the transmit idle state, issue `symrdf deactivate -rdfa_pace -symforce` from the R1 side.

For example, to activate group-level and device-level write pacing simultaneously for the `ConsisGrp` CG, type:

```
symrdf -cg ConsisGrp activate -rdfa_pace
```

To exempt `DEV001` in the `prod` group from both group-level and device-level write pacing, type:

```
symrdf -g prod -sid 55 -rdfg 76 -rdfa_pace_exempt DEV001
```

To deactivate both group-level and device-level write pacing on the devices in `DeviceFile2`, type:

```
symrdf -file DeviceFile2 -sid 55 -rdfg 2 deactivate -rdfa_pace
```

Cleanup operation for SRDF/A

Solutions Enabler provides the `msc_cleanup` command for devices operating in SRDF/A mode with consistency enabled for Multi-Session Consistency (MSC). The command may be necessary in certain fault scenarios where all delta sets of a transition have not been fully applied or discarded. The command can be executed by composite group from the R1 or R2 site or by SRDF group from the R2 site. The command maintains dependent write consistency by discarding any incomplete data and committing completed data to the R2 site.

To perform a cleanup operation on a composite group (`mycg`) operating in SRDF/A mode, type:

```
symrdf -cg mycg msc_cleanup
```

Enable and disable consistency protection for SRDF/A devices

The `enable` action enables consistency protection for devices in SRDF/Asynchronous mode by device group or device list. If data cannot be copied from the R1 to the R2, all devices in the group will be made not ready on the links to preserve R2 data consistency.

For example, to enable consistency protection for SRDF/A pairs in device group `prod`, type:

```
symrdf -g prod enable
```

To enable consistency protection for SRDF/A pairs listed in device file `devfile1`, type:

```
symrdf -file devfile1 -sid 123 -rdfg 10 enable
```

The `disable` action disables consistency protection for devices in SRDF/Asynchronous mode by device group or device list. If data cannot be copied from the R1 to the R2, then only the devices in the group that are experiencing problems will be made not ready on the links. The device state for any remaining devices in the group will remain the same.

For example, to disable consistency protection for SRDF/A pairs in device group `prod`, type:

```
symrdf -g prod disable
```

To disable consistency protection for SRDF/A pairs listed in device file `devfile1`, type:

```
symrdf -file devfile1 -sid -rdfg 10 disable
```

Note: To enable consistency protection for SRDF/A pairs listed in a composite group (`-cg`), refer to "SRDF consistency group operations" on page 202.

Establish (full)

You must perform a full establish on SRDF pairs only when you are initially setting up SRDF pairs, or when your R2 member of an SRDF pair is either fully invalid, or has been replaced.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

When the establish control operation has successfully completed and the device pair has fully synchronized, the SRDF pairs will contain identical data. You can use `verify` to confirm that the SRDF pairs are in the Synchronized state and remote mirroring is resumed.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issued the `establish -full` operation on a device group, composite group, and device file:

```
symrdf -g DgName establish -full
symrdf -cg CgName establish -full
symrdf -f[ile] FileName establish -full
```

For details about defining a device file, refer to "Displaying SRDF information" on page 86.

For example, to initiate an establish for all the SRDF pairs in the device group `prod`, type:

```
symrdf -g prod establish -full
```

To initiate an establish for one SRDF pair with logical device `DEV001` in the device group `prod`, type:

```
symrdf -g prod establish -full DEV001
```

To initiate an establish for a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod establish -full DEV001 DEV002 DEV003
```

Note: The R2 may be set to read/write disabled (not ready) if `SYMAPI_RDF_RW_DISABLE_R2=ENABLE` is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Figure 4 illustrates the establishing of an SRDF pair. The SRDF pair consists of the source (R1) device that is mirrored to the target (R2) device.

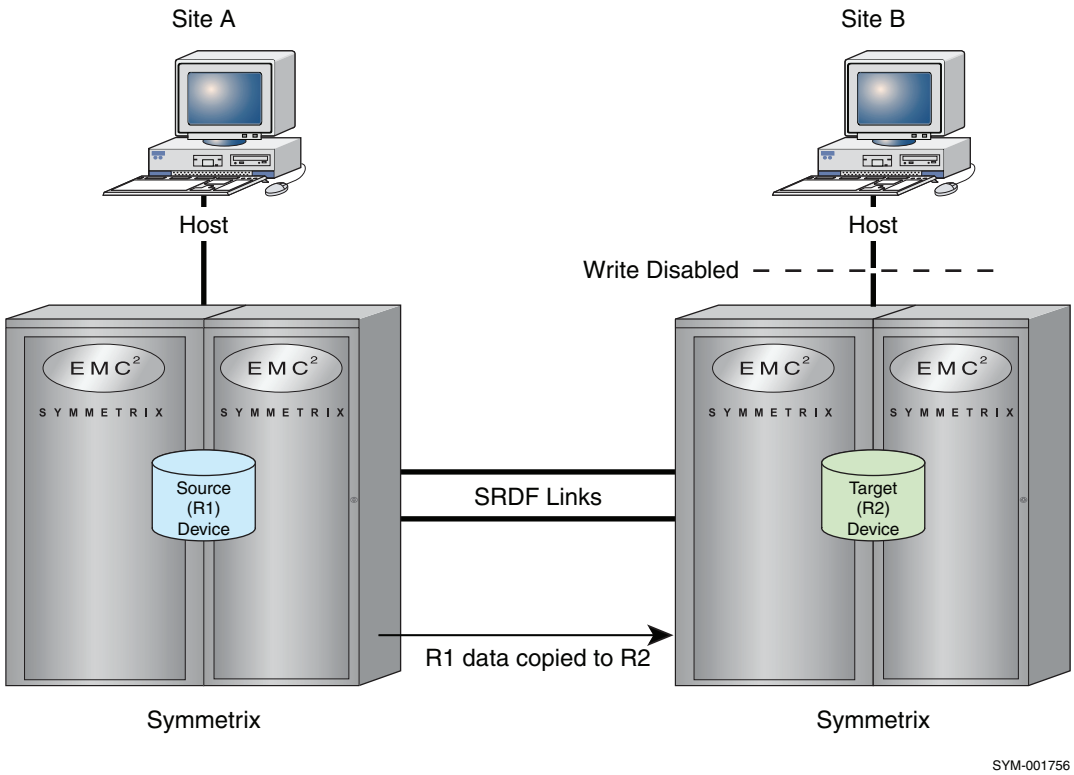


Figure 4 Establishing an SRDF pair

When a full establish is initiated for each specified SRDF pair in a device group, the following occurs:

- The target (R2) device is write disabled to its local host I/O.
- Traffic is suspended on the SRDF links.
- All the tracks on the target (R2) device are marked invalid.
- All tracks on the R2 side are refreshed by the R1 source side. The track tables are merged between the R1 and R2 side.
- Traffic is resumed on the SRDF links.

The SRDF pair is in the Synchronized state when the source (R1) device and the target (R2) device contain identical data.

Establish (incremental)

Incrementally establishing an SRDF pair (Figure 5 on page 63) accomplishes the same thing as the establish process, with a major time-saving exception: the source (R1) device copies to the target (R2) device only the new data that was updated on the source (R1) device while the SRDF pair was split. Additionally, any data that was modified on the target (R2) device will be refreshed from the corresponding tracks on the source (R1) side.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

When the establish control operation has successfully completed and the SRDF device pair has synchronized, the SRDF pairs will contain identical data.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the incremental establish control operation on a device group, composite group, and device file:

```
symrdf -g DgName establish
symrdf -cg CgName establish
symrdf -f[ile] FileName establish
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to initiate an incremental establish on all SRDF pairs in the `prod` device group, type:

```
symrdf -g prod establish
```

To initiate an incremental establish on one SRDF pair with logical device `DEV001` in the `prod` device group, type:

```
symrdf -g prod establish DEV001
```

To initiate an incremental establish for a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod establish DEV001 DEV002 DEV003
```

Note: R2 may be set to read/write disabled (not ready) if SYMAPI_RDF_RW_DISABLE_R2=ENABLE is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Figure 5 illustrates the incremental establishing of an SRDF pair. The SRDF pair consists of the source (R1) device that is mirrored to the target (R2) device.

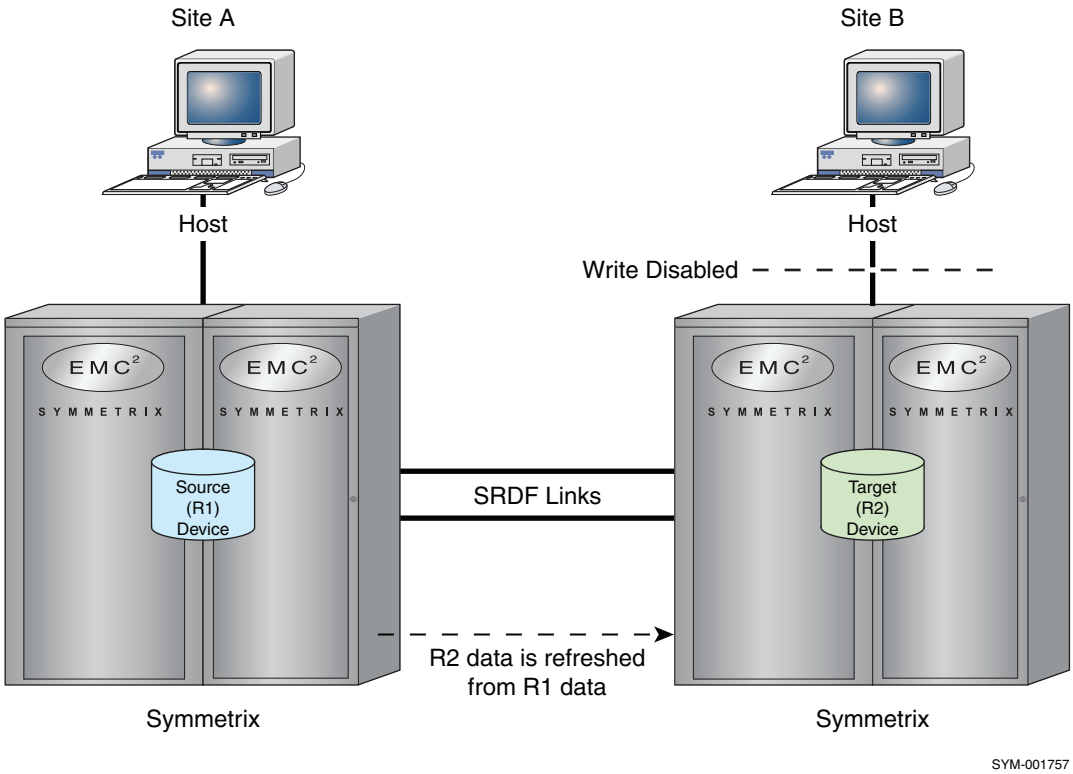


Figure 5 Incremental establish of an SRDF pair

When an incremental establish is initiated for each specified SRDF pair in a device group, the following occurs:

- The target (R2) device is write disabled to its local host I/O.
- Traffic is suspended on the SRDF links.
- The invalid tracks on the target (R2) device are refreshed from the changed tracks of the source (R1) device.
- The track tables are merged between the source (R1) device and the target (R2) device.
- Traffic is resumed on the SRDF links.

The SRDF pair is in the Synchronized state when the source (R1) device and the target (R2) device contain identical data.

Failback

A failback, or source (R1) device takeover, is performed when you are ready to resume normal SRDF operations by initiating read/write operations on the source (R1) devices, and stop read/write operations on the target (R2) devices. The target (R2) devices become read-only to their local hosts while the source (R1) devices are read/write enabled to their local hosts.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. The failback control operation can be performed by device group, composite group, or device file:

```
symrdf -g DgName failback
symrdf -cg CgName failback
symrdf -f[ile] FileName failback
```

Note: For details about defining a device file, refer to "Displaying SRDF information" on page 86.

For example, to initiate a failback on all the SRDF pairs in the `prod` device group, type:

```
symrdf -g prod failback
```

To initiate a failback on one SRDF pair, `DEV001`, in the `prod` device group, type:

```
symrdf -g prod failback DEV001
```

To initiate a failback on a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod failback DEV001 DEV002 DEV003
```

Note: The R2 may be set to read/write disabled (not ready) if `SYMAPI_RDF_RW_DISABLE_R2=ENABLE` is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Figure 6 illustrates the failback of an SRDF pair. The SRDF pair consists of the source (R1) device which is mirrored to the target (R2) device.

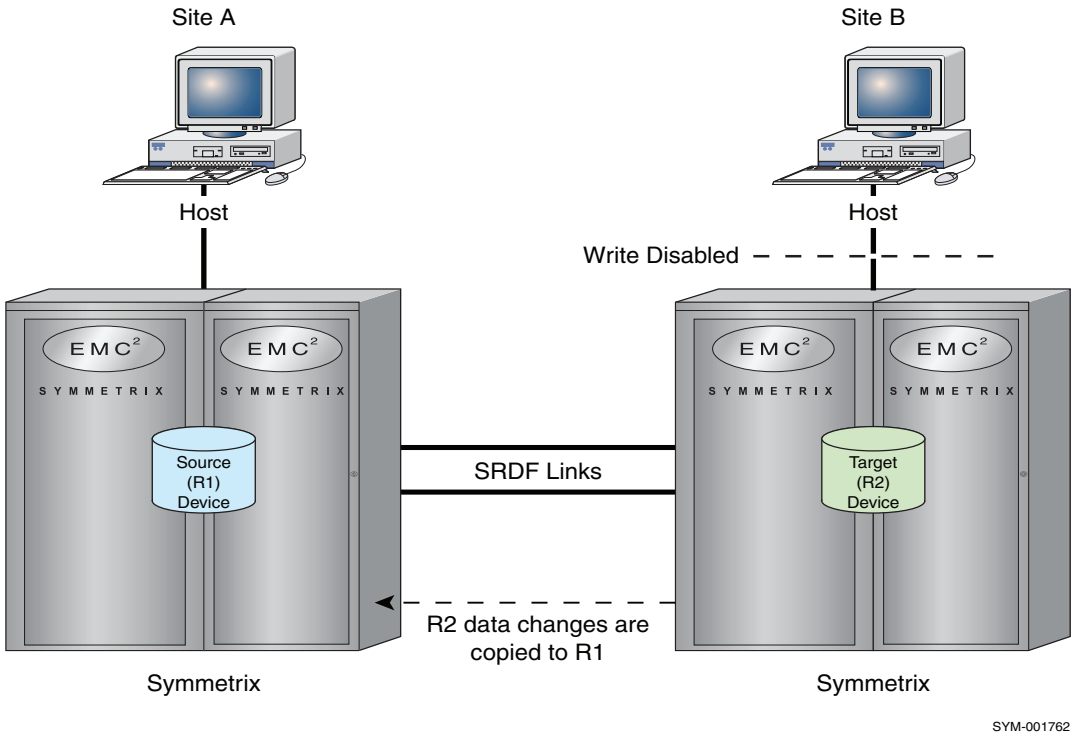


Figure 6 Failback of an SRDF device

When a failback is initiated for each specified SRDF pair in a device group, the following occurs:

- The target (R2) device is write disabled to its local hosts.
- Traffic is suspended on the SRDF links.
- If the target side is operational, and there are invalid remote (R2) tracks on the source side (and the force option is specified), the invalid R1 source tracks are marked to refresh from the target side.
- The invalid tracks on the source (R1) side are refreshed from the target R2 side. The track tables are merged between the R1 and R2 sides.
- Traffic is resumed on the SRDF links.
- The source (R1) device is read/write enabled to its local hosts.

Note: If the target is reachable, use the `-force` option to mark the changed tracks on the source side to refresh from the target.

Failover

In a period of scheduled downtime for maintenance, or after a serious system problem that has rendered either the host or Symmetrix array containing the source (R1) devices unreachable, no read/write operations can occur on the source (R1) device. In this situation, the *failover* operation should be initiated to make the target (R2) devices read/write enabled to their local hosts.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `failover` operation on a device group, composite group, and device file:

```
symrdf -g DgName failover
symrdf -cg CgName failover
symrdf -f[ile] FileName failover
```

Note: For details about defining a device file, refer to "Displaying SRDF information" on page 86.

For example, to perform a failover on all the pairs in the `prod` device group, type:

```
symrdf -g prod failover
```

To perform a failover on one SRDF pair with device `DEV001` in the `prod` device group, type:

```
symrdf -g prod failover DEV001
```

To perform a failover on a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod failover DEV001 DEV002 DEV003
```

Figure 7 illustrates the failover of an SRDF pair. The SRDF pair consists of the source (R1) device, which is mirrored to the target (R2) device.

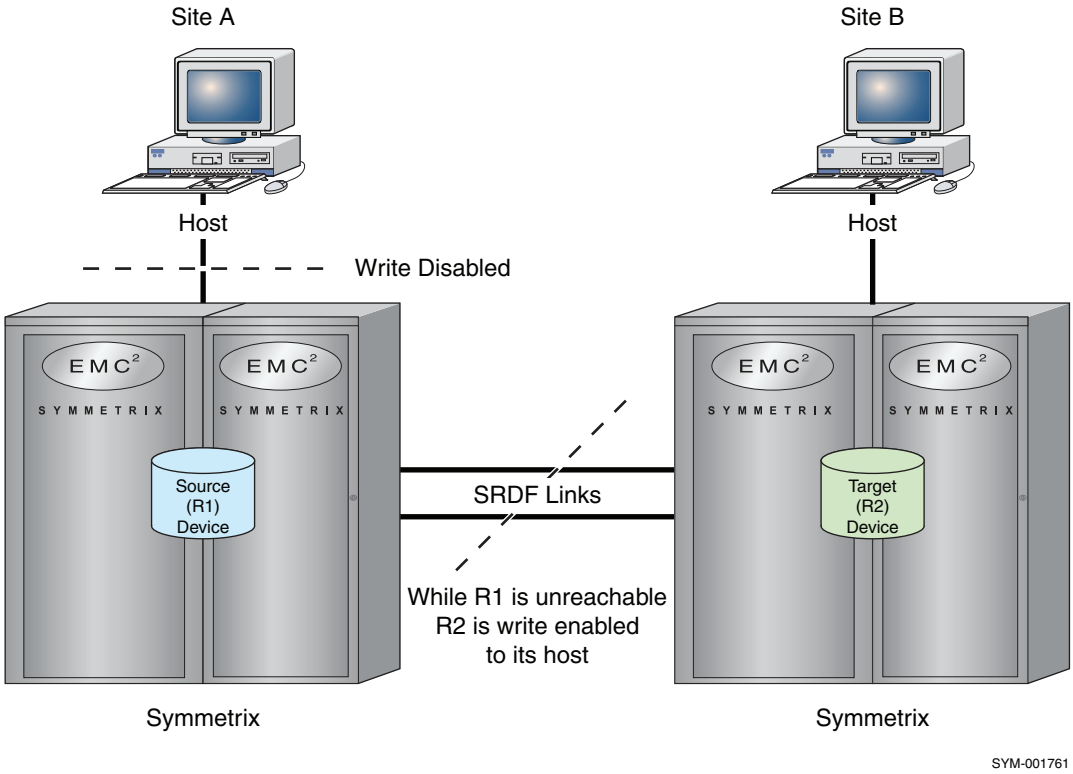


Figure 7 Failover of an SRDF device

When a failover is performed for each specified SRDF pair in a device group, the following occurs:

- If the source (R1) device is operational, the SRDF links are suspended.
- If the source side is operational, the source (R1) device is write disabled to its local hosts.
- The target (R2) device is read/write enabled to its local hosts.

Invalidate R1 tracks

The invalidate R1 action invalidates all tracks on the source (R1) side so that they can be copied over from the target (R2) side.

You can issues the `invalidate r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName invalidate r1
symrdf -cg CgName invalidate r1
symrdf -f[ile] FileName invalidate r1
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to invalidate the source (R1) devices in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod invalidate r1
```

To invalidate the source (R1) device in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod invalidate r1 DEV007
```

To invalidate the source (R1) device for a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod invalidate r1 DEV002 DEV003 DEV007
```

To invoke this operation, the SRDF pairs at the source must already be Suspended and write disabled (not ready).

Note: Using the `-nowd` option with the `invalidate r1` command allows you to bypass the validation check to ensure that the target of operation is write disabled to the host.

Invalidate R2 tracks

The invalidate R2 mirror action invalidates all tracks on the target (R2) side so that they can be copied over from the source (R1) side.

You can issue the `invalidate R2` operation on a device group, composite group, and device file:

```
symrdf -g DgName invalidate r2
symrdf -cg CgName invalidate r2
symrdf -f[ile] FileName invalidate r2
```

Note: For details about defining a device file, refer to "Displaying SRDF information" on page 86.

For example, to invalidate the target (R2) devices in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod invalidate r2
```

To invalidate the target (R2) device in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod invalidate r2 DEV007
```

Note: Using the `-nowd` option with the invalidate r2 command allows you to bypass the validation check to ensure that the target of operation is write disabled to the host.

Make R1 not ready

The make R1 mirror not ready action sets the source (R1) devices to not ready for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for the control operations. You issue the `not_ready r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName not_ready r1
symrdf -cg CgName not_ready r1
symrdf -f[ile] FileName not_ready r1
```

Note: "Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to make the source (R1) devices not ready in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod not_ready r1
```

To make the source (R1) device not ready in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod not_ready r1 DEV007
```

To make the source (R1) device not ready in a list of SRDF pairs, `DEV007`, in device group `prod`, type:

```
symrdf -g prod not_ready r1 DEV002 DEV003 DEV007
```

Make R1 ready

The make R1 mirror ready action sets the source (R1) devices to ready for their local hosts. This operation is particularly helpful when all SRDF links are lost and the devices are operating in domino mode.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can perform the `ready r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName ready r1
symrdf -cg CgName ready r1
symrdf -f[ile] FileName ready r1
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to make the source (R1) device ready in all the pairs in device group `prod`, type:

```
symrdf -g prod ready r1
```

To make the source (R1) device ready in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod ready r1 DEV007
```

To make the source (R1) device ready in a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod ready r1 DEV002 DEV003 DEV007
```

Make R2 not ready

The make R2 mirror not ready action sets the target (R2) devices to not ready for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `not_ready r2` operation on a device group, composite group, and device file:

```
symrdf -g DgName not_ready r2
symrdf -cg CgName not_ready r2
symrdf -f[ile] FileName not_ready r2
```

"Displaying SRDF information" on page 86 provides detail about defining a device file.

For example, to make the target (R2) devices not ready in all pairs in device group `prod`, type:

```
symrdf -g prod not_ready r2
```

To make the target (R2) device in one pair not ready, `DEV007`, in device group `prod`, type:

```
symrdf -g prod not_ready r2 DEV007
```

To make the target (R2) device not ready in a list of pairs in device group `prod`, type:

```
symrdf -g prod not_ready r2 DEV002 DEV003 DEV007
```

Make R2 ready

The make R2 mirror ready action sets the target (R2) devices to ready for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `ready r2` operation on a device group, composite group, and device file:

```
symrdf -g DgName ready r2
symrdf -cg CgName ready r2
symrdf -f[ile] FileName ready r2
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to make the target (R2) devices ready in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod ready r2
```

To make the target (R2) device ready in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod ready r2 DEV007
```

To make the source (R2) device ready in a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod ready r2 DEV002 DEV003 DEV007
```

Merge track tables

The merge track tables action merges the track tables between the source (R1) and the target (R2) devices. This option allows for the comparison of track tables on SRDF device pairs in a device group and may be used to compare the track tables between SRDF device pairs that have been split and re-established.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `merge` operation on a device group, composite group, and device file:

```
symrdf -g DgName merge
symrdf -cg CgName merge
symrdf -f[ile] FileName merge
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to merge the track tables of all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod merge
```

To merge the track table of one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod merge DEV007
```

To merge the track table of a list SRDF pairs in device group `prod`, type:

```
symrdf -g prod merge DEV002 DEV003 DEV007
```

Move one-half of an SRDF pair

The `half_movepair` command moves one side of a dynamic pair from one group to another. You can only move devices from R1 to R1 or from R2 to R2.

For example, to move one-half of the SRDF pairing of SRDF group 10 to a new SRDF group 15, type:

```
symrdf half_movepair -sid 123 -file devicefile -rdfg 10 -new_rdfg 15
```

This command moves the first device listed in each line of the device file to the new SRDF group.

"Moving dynamic SRDF pairs" on page 107 also provides information on the `symrdf movepair` command for device files.

Move SRDF device pairs

The `movepair` command moves devices from one SRDF group to another without requiring a full resynchronization. You can only move devices from R1 to R1, or from R2 to R2.

For example, to move pairs in a file from SRDF group 10 to SRDF group 15, type:

```
symrdf movepair -sid 123 -file devicefile -rdfg 10 -new_rdfg 15
```

This moves the first line of the device file to the new SRDF group, and moves the second device listed in each line of the remote SRDF group that is paired with the new SRDF group.

The `movepair` command can also be executed on device groups `(-g)` instead of specifying the device text file.

Read/write disable target device

The read /write disable R2 action blocks both reads from and writes to the target (R2) devices from their local host. This option enables a user to set a device to the not ready state on the R2 side by making the device not ready on the RA.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `rw_disable r2` operation on a device group, composite group, and device file:

```
symrdf -g DgName rw_disable r2
symrdf -cg CgName rw_disable r2
symrdf -f[ile] FileName rw_disable r2 -rdfg2
```

"Displaying SRDF information" on page 86 for details on the symrdf movepair command for device files.

For example, to read/write disable all the target (R2) mirrors in the SRDF pairs in a device group `prod`, type:

```
symrdf -g prod rw_disable r2
```

To read/write disable the target (R2) mirror in the SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod rw_disable r2 DEV007
```

To read/write disable the target (R2) mirror in a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod rw_disable r2 DEV002 DEV003 DEV007
```

Refresh R1

The refresh R1 mirror action marks any changed tracks on the source (R1) side to refresh from the R2 side.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `refresh r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName refresh r1
symrdf -cg CgName refresh r1
symrdf -f[ile] FileName refresh r1
```

For example, to refresh all the source (R1) devices in all the SRDF pairs in the device group `prod`, type:

```
symrdf -g prod refresh r1
```

To refresh the source (R1) device in the SRDF pair, `DEV007`, in the device group `prod`, type:

```
symrdf -g prod refresh r1 DEV007
```

To refresh the source (R1) device in the list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod refresh r1 DEV002 DEV003 DEV007
```

Refresh R2

The refresh R2 mirror action marks any changed tracks on the target (R2) side to refresh from the R1 side.

You can issue the `refresh r2` operation on a device group, composite group, and device file:

```
symrdf -g DgName refresh r2
symrdf -cg CgName refresh r2
symrdf -f[ile] FileName refresh r2
```

For example, to refresh the target (R2) devices in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod refresh r2
```

To refresh the target (R2) device in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod refresh r2 DEV007
```

To refresh the target (R2) device for a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod refresh r2 DEV002 DEV003 DEV007
```

Restore (full)

The full restore operation differs from the establish operations in that the entire contents of the target (R2) device is copied to the source (R1) device. After the restore control operation is completed, the pairs synchronize. You can use verify to confirm that the SRDF pairs are in the Synchronized state.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. The full restore operation can be performed by device group, composite group, or device file:

```
symrdf -g DgName restore -full
symrdf -cg CgName restore -full
symrdf -f[ile] FileName restore -full
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to initiate a full restore on all SRDF pairs in the `prod` device group, type:

```
symrdf -g prod restore -full
```

To initiate a full restore on one SRDF pair with logical device `DEV001` in the `prod` device group, type:

```
symrdf -g prod restore -full DEV001
```

To initiate a full restore on a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod restore -full DEV001 DEV002 DEV003
```

Note: R2 may be set to read/write disabled (not ready) if `SYMAPI_RDF_RW_DISABLE_R2=ENABLE` is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Figure 8 illustrates restoring an SRDF pair. The SRDF pair consists of the source (R1) device, mirrored to the target (R2) device.

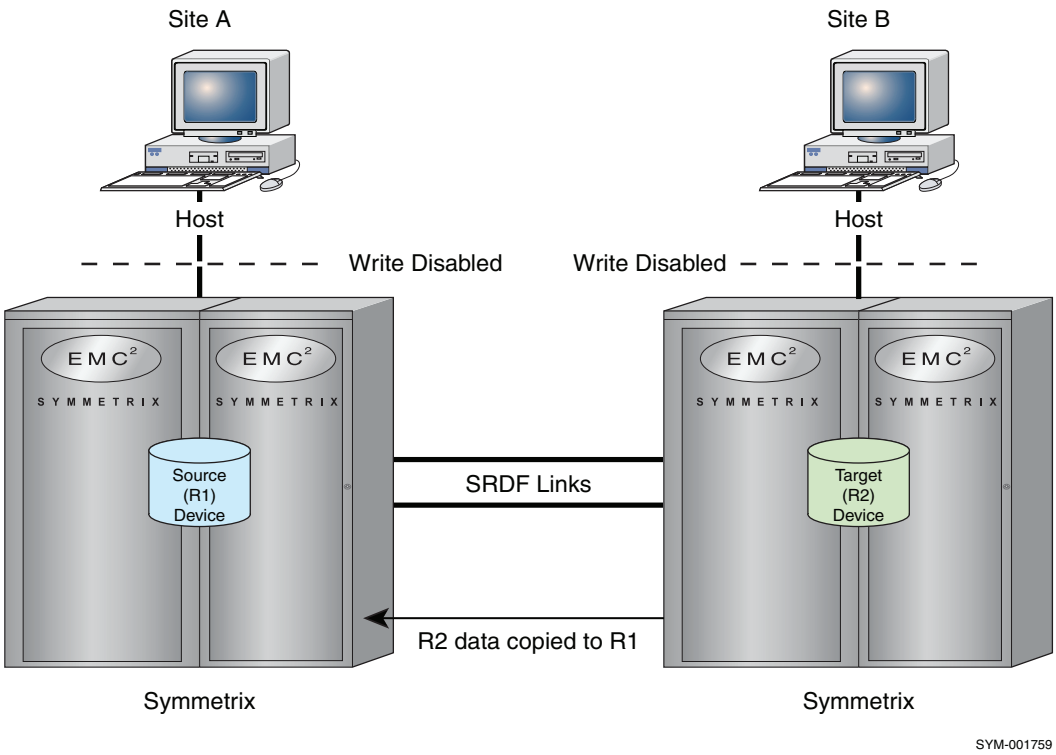


Figure 8 Restoring an SRDF device

When a restore is initiated for each specified SRDF pair in a device group, the following occurs:

- The source (R1) device is write disabled to its local hosts.
- The target (R2) device is write disabled to its local hosts.
- Traffic is suspended on the SRDF links.
- All tracks on the source (R1) device are marked as invalid.
- All R1 tracks are refreshed from the R2 side. The track tables are merged between the R1 and R2 side.
- Traffic is resumed on the SRDF links.
- The source (R1) device is read/write enabled to its local hosts.

The restoration process is complete when the source (R1) and target (R2) device contain identical data. After the restore is complete, the SRDF pair is in the Synchronized state.

Restore (incremental)

The incremental restore process accomplishes the same thing as the restore process with a major time-saving exception: the target (R2) device copies to the source (R1) device only the new data that was updated on the target (R2) device while the SRDF pair was split. Any changed tracks on the source (R1) device are refreshed from the corresponding tracks on the target (R2) device. After the restore control operation has successfully completed, the SRDF pairs will synchronize. You can use verify to confirm that the SRDF pairs are in the Synchronized state.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

This process is useful if the results from running a new application on the target (R2) device were desirable, and the user wants to move the changed data and the new application to the source (R1) device.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. The incremental restore operation can be performed by device group, composite group, or device file:

```
symrdf -g DgName restore
symrdf -cg CgName restore
symrdf -f[ile] FileName restore
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to initiate an incremental restore on all SRDF pairs in the `prod` device group, type:

```
symrdf -g prod restore
```

To initiate an incremental restore on one SRDF pair with logical device `DEV001` in the `prod` device group, type:

```
symrdf -g prod restore DEV001
```

To initiate an incremental restore for a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod restore DEV001 DEV002 DEV003
```

Note: R2 may be set to read/write disabled (not ready) if `SYMAPI_RDF_RW_DISABLE_R2=ENABLE` is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

Figure 9 illustrates the incremental restore of an SRDF pair. The SRDF pair consists of the source (R1) device, which is mirrored to the target (R2) device.

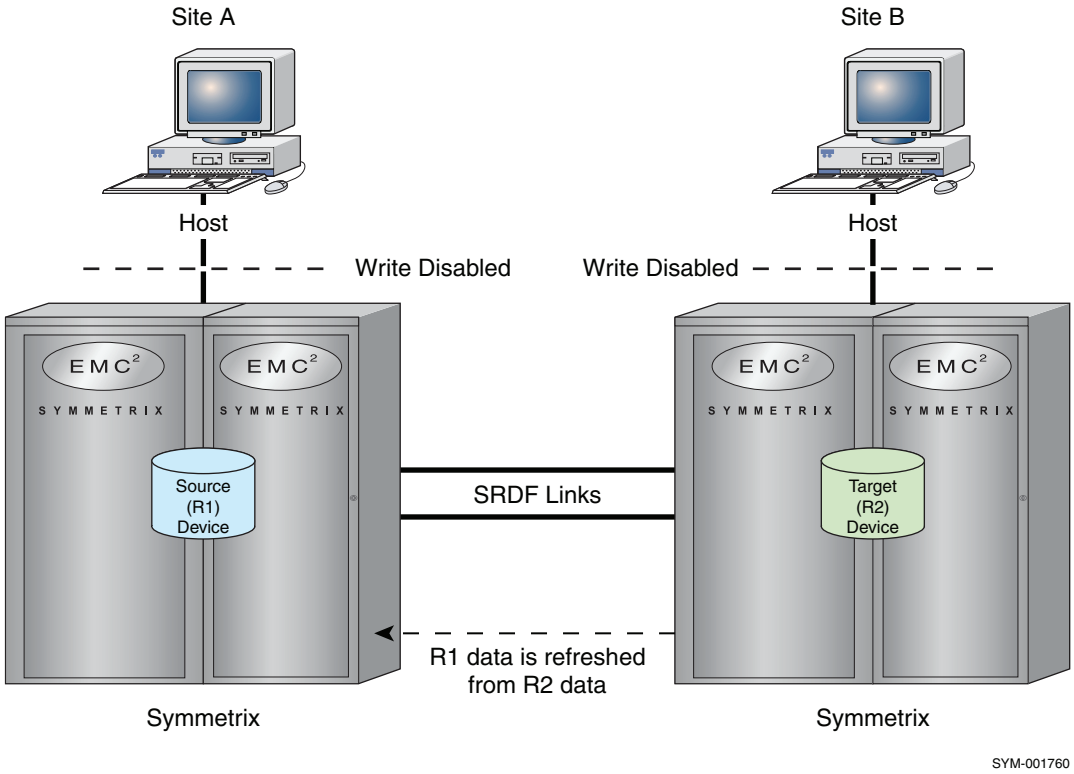


Figure 9 Incrementally restoring an SRDF device

When an incremental restore is initiated for each specified SRDF pair in a device group, the following occurs:

- The source (R1) device is write disabled to its local hosts.
- The target (R2) device is write disabled to its local hosts.
- Traffic is suspended on the SRDF links.
- The invalid tracks on the source (R1) device are refreshed from the changed tracks on the target (R2) side. The track tables are merged between the R1 and R2 side.
- Traffic is resumed on the SRDF links.
- The source (R1) device is read/write enabled to its local hosts.

The SRDF pair is in the Synchronized state when the source (R1) device and the target (R2) device contain identical data.

Resume I/O on links

The `resume` action argument resumes I/O traffic on the SRDF links for all remotely mirrored SRDF pairs in a group or device file.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can execute the `resume` control operation on device groups, composite groups, and device files:

```
symrdf -g DgName resume
symrdf -cg CgName resume
symrdf -f[ile] FileName resume
```

For example, to resume the SRDF links between all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod resume
```

To resume the SRDF links between one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod resume DEV007
```

Note: If you do not specify `-force` when the merge track table is required, the `resume` operation is rejected.

To resume the SRDF links (between the pairs) on a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod resume DEV002 DEV003 DEV007
```

Split

You must split SRDF pairs when you require read and write access to the target (R2) side of one or more devices in a device group, composite group, or device file.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `split` operation on a device group, composite group, and device file:

```
symrdf -g DgName split
symrdf -cg CgName split
symrdf -f[ile] FileName split
```

"Displaying SRDF information" on page 86 provides details about defining a device file.

For example, to perform a split on all the SRDF pairs in the `prod` device group, type:

```
symrdf -g prod split
```

To perform a split on one SRDF pair with logical device `DEV001` in the `prod` group, type:

```
symrdf -g prod split DEV001
```

To initiate a split to a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod split DEV001 DEV002 DEV003
```

Figure 10 illustrates the splitting of a SRDF pair. The SRDF pair consists of the source (R1) device, which is remotely mirrored to the target (R2) device.

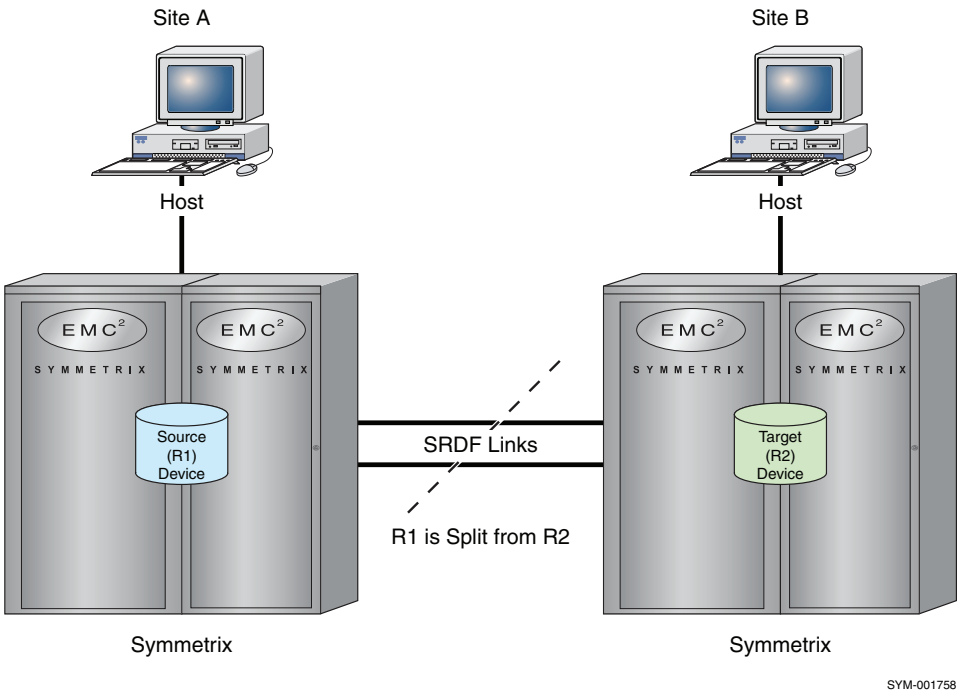


Figure 10 Splitting an SRDF pair

When a split is performed for each specified SRDF pair in a device group, the following occurs:

- Traffic is suspended on the SRDF links.
- The target (R2) device is read/write enabled to its local hosts.
- After the target (R2) device is split from the source (R1) device, the SRDF pair is in the Split state.

Note: If you do not specify `-force` when the device pairs are in domino mode or adaptive copy mode, the `split` operation is rejected.

Splits impacting databases

If the `split` action impacts the access integrity of a database, additional actions such as freezing the database to user access may be necessary. You can use the `freeze` action in conjunction with the TimeFinder or `split` operation. The freeze action suspends the database updates being written to disk.

Using the `symioctl` command, you can invoke I/O control operations to freeze access to a specified relational database or database objects.

Note: For access to the specified database, set SYMCLI_RDB_CONNECT to your username and password.

Freeze

To freeze all I/O access to a specified relational database, you use the following command:

```
symioctl freeze -type DbType Object Object
```

SQL Server allows some or all databases to be specified. Oracle and Informix allow you to freeze or thaw an entire DB system.

If you have set the connection environment variables, you just need to type:

```
symioctl freeze Object Object
```

For example, to freeze databases HR and Payroll, type:

```
symioctl freeze HR Payroll
```

Thaw

Once the freeze action is completed, the split may proceed. When the split operation completes, a `symioctl thaw` command must be sent to resume full I/O access to the database instance. For example:

```
symioctl thaw
```

Hot backup control

For Oracle only, you can perform hot backup control on a list of tablespace objects, which must be performed before and after a freeze/thaw command. The steps required to split a group of SRDF pairs follows:

1. Issue the `symioctl begin backup` command.
2. Issue the `symioctl freeze` command.
3. Split the SRDF pairs. This may involve several steps depending on your environment.
4. Issue the `symioctl thaw` command.
5. Issue the `symioctl end backup` command.

Consistency groups

For consistency group split operations, refer to "SRDF consistency group operations" on page 202.

Suspend I/O on links

The `suspend` action suspends I/O traffic on the SRDF links for all remotely mirrored SRDF pairs in the group or device file.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `suspend` operation on a device group, composite group, and device file:

```
symrdf -g DgName suspend [-immediate | -cons_exempt]
symrdf -cg CgName suspend [-immediate | -cons_exempt]
symrdf -f[ile] FileName suspend [-immediate | -cons_exempt]
```

For example, to suspend the SRDF links between all the pairs in device group `prod`, type:

```
symrdf -g prod suspend
```

To suspend the SRDF links between one pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod suspend DEV007
```

To suspend the SRDF links (between the pairs) on a list of pairs in device group `prod`, type:

```
symrdf -g prod suspend DEV002 DEV003 DEV007
```

Note: When used with the consistency exempt (-cons_exempt) option, allows devices to be suspended without affecting the state of the SRDF/A session or requiring that other devices in the session be suspended.

When the suspend has completed successfully, the devices are suspended on the SRDF links and their link status is set to not ready (NR).

Note: The `suspend` operation is rejected when the device is in domino mode.

Suspend/resume timestamp

You can show SRDF link status changes from read/write to not ready and not ready to read/write. The suspend/resume link status information is displayed in the SRDF Information output section for the `symdev`, `sympd`, and `symdg show` commands. The timestamp is relative to the clock on the host where the command was issued and is reported for each SRDF mirror on both the R1 and R2 mirrors. This feature is available with Symmetrix Enginuity 5773.150 and higher.

Note: This timestamp is not associated with the R2 data for SRDF/A.

Swap one-half of an SRDF pair

You can swap one-half of a designated SRDF pair as specified in a device file, device group, or composite group. The `half_swap` command swaps the personality of one half of an SRDF relationship. It changes an R1 mirror to an R2 mirror or an R2 mirror to an R1 mirror.

For example, to swap the R1 designation of the associated BCV RDF1 pairs in device group `prod`, and refresh the data on the current R1 side, type:

```
symrdf -g Prod -bcv half_swap -refresh R1
```

For more information, refer to "Issuing a dynamic half-swap" on page 109.

Swap SRDF pairs

You can swap the personality of SRDF device pairs. The source (R1) device becomes the target (R2) device and the target (R2) device becomes the source (R1) device. For example, to swap the R1 designation of the associated BCV RDF1 pairs in device group `prod`, and refresh the data on the current R1 side, type:

```
symrdf -g Prod -bcv swap -refresh R1
```

For detailed information on the `symrdf swap` command for device files, device groups, or composite groups refer to "Issuing a dynamic R1/R2 swap" on page 108.

Note: A swap is not allowed if the R1 device (which becomes the R2) is currently a target for a TimeFinder/Snap or TimeFinder/Clone emulation. A device may not have two sources for data (in this case, the R1 and the emulation source). This is not allowed even if the emulation session has already completed copying the data.

Update R1 mirror

While the target (R2) device is still operational (Write Enabled to its local host), an incremental data copy from the target (R2) device to the source (R1) device can be initiated in order to *update the R1 mirror* with changed tracks from the target (R2) device.

Note: When the `symrdf` command is initiated, device external locks are set on all SRDF devices you are about to establish. Device external locks are then automatically released when the control operation completes. For information on listing device external locks, or releasing device locks, refer to "Device external locks" on page 42.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `update` operation on a device group, composite group, and device file:

```
symrdf -g DgName update
symrdf -cg CgName update
symrdf -f[ile] FileName update
```

For details about defining a device file, refer to "Displaying SRDF information" on page 86.

For example, to initiate an update of all the source (R1) devices in the SRDF pairs, for device group `prod`, type:

```
symrdf -g prod update
```

To initiate an update of the source (R1) device in the SRDF pair with logical device `DEV001` in device group `prod`, type:

```
symrdf -g prod update DEV001
```

To initiate an update on a list of SRDF pairs in the device group `prod`, type:

```
symrdf -g prod update DEV001 DEV002 DEV003
```

Figure 11 illustrates the update of an SRDF pair. The SRDF pair consists of the source (R1) device which is mirrored to the target (R2) device.

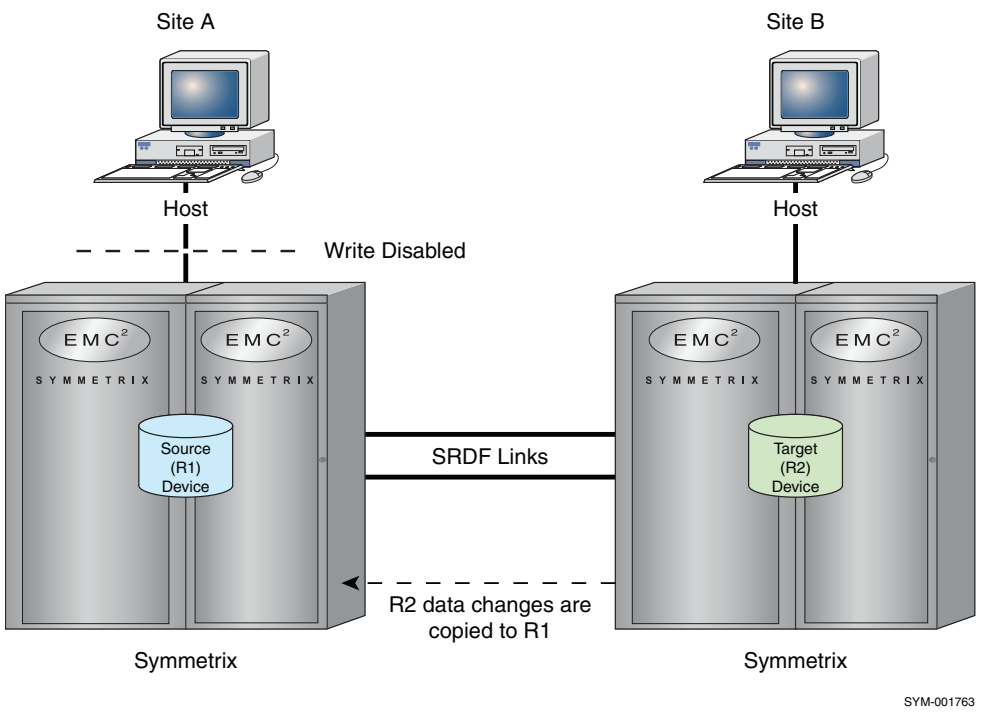


Figure 11 Update of SRDF device track tables

An update is initiated for each specified SRDF pair in a device group as follows:

- The SRDF (R1 to R2) links are suspended when the SRDF links are up.
- If there are invalid remote (R2) tracks on the source side and the force option was specified, tracks that were changed on the source devices are marked to refresh from the target side.
- The invalid tracks on the source (R1) side are refreshed from the target R2 side. The track tables are merged between the R1 and R2 sides.
- Traffic is resumed on the SRDF links.

When the update has completed successfully, the pairs is in the R1 Updated state.

IMPORTANT

When you perform an update while the SRDF pair is Suspended and are not ready at the source, the SRDF pair types an Invalid state as the update completes. To resolve this condition, issue `rw_enable r1` for the SRDF pairs to become Synchronized.

Continuous R1 updates

You can perform continuous updates with one command (`update -until #`) for situations when you have or want I/O to continue from the remote host and periodically update an inactive R1 device over an extended period of time.

The until (`-until`) option when used with the `update` argument checks the number of invalid tracks that are allowed to build up from the active R2 local I/O before another update (R2 to R1 copy) is triggered. The update sequence loops until the invalid track count is less than the number specified for the `-until` value.

Note that these update sequences start with an immediate update once this command is started as follows:

1. Update R1 mirror.
2. Changed tracks are built on R2.
3. Check invalid track count.

Note: If the invalid track count is less than the number of tracks specified for the `-until` value, the command exits, otherwise, the above sequence of operations for update R1 mirror is retriggered until the threshold is reached.

For example, to update the R1 mirror when track changes are in excess of 1000 on the R2, type:

```
symrdf -g prod update -until 1000
```

In this example, the R1 mirror will be continuously updated until the number of tracks to be copied is below 1000.

Write disable R1

The write disable R1 action write disables the source (R1) devices for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `write_ disable r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName write_disable r1
symrdf -cg CgName write_disable r1
symrdf -f[ile] FileName write_disable r1
```

For example, to write disable all the source (R1) mirrors in the SRDF pairs in device group `prod`, type:

```
symrdf -g prod write_disable r1
```

To write disable the source (R1) mirror in the SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod write_disable r1 DEV007
```

To write disable the source (R1) mirror in a list of SRDF pairs, (`DEV002`, `DEV003`, `DEV007`) in device group `prod`, type:

```
symrdf -g prod write_disable r1 DEV002 DEV003 DEV007
```

Write disable R2

The write disable R2 mirror action argument write disables the target (R2) devices to their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. The write disable R2 mirror control operation can be performed by device group, composite group, or device file:

```
symrdf -g DgName write_disable r2
symrdf -cg CgName write_disable r2
symrdf -f[ile] FileName write_disable r2
```

For example, to write disable all the target (R2) mirrors in the SRDF pairs in device group `prod`, type:

```
symrdf -g prod write_disable r2
```

To write disable the target (R2) mirror in the SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod write_disable r2 DEV007
```

To write disable the target (R2) mirror in a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod write_disable r2 DEV002 DEV003 DEV007
```

Write enable R1

The read/write enable R1 action enables the source (R1) devices for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `rw_enable r1` operation on a device group, composite group, and device file:

```
symrdf -g DgName rw_enable r1
symrdf -cg CgName rw_enable r1
symrdf -f[ile] FileName rw_enable r1
```

For example, to read/write enable all the source (R1) mirrors in all the SRDF pairs in device group `prod`, type:

```
symrdf -g prod rw_enable r1
```

To read/write enable the source (R1) mirrors in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod rw_enable r1 DEV007
```

Write enable R2

The read/write enable R2 action write enables the target (R2) devices for their local hosts.

Table 55 on page 498 lists the applicable SRDF pair states for this operation. You can issue the `rw_enable R2` operation on a device group, composite group, and device file:

```
symrdf -g DgName rw_enable r2
symrdf -cg CgName rw_enable r2
symrdf -f[ile] FileName rw_enable r2
```

For example, to read/write enable all the target (R2) mirrors in the SRDF pairs in device group `prod`, type:

```
symrdf -g prod rw_enable r2
```

To read/write enable the target (R2) mirror in one SRDF pair, `DEV007`, in device group `prod`, type:

```
symrdf -g prod rw_enable r2 DEV007
```

To read/write enable the target (R2) mirror of a list of SRDF pairs in device group `prod`, type:

```
symrdf -g prod rw_enable r2 DEV002 DEV003 DEV007
```

Displaying SRDF information

By issuing various SYMCLI commands, you can view configuration and status information about the SRDF devices on your Symmetrix arrays. "SRDF Control Operations" on page 47 provides example output of these commands.

Finding Symmetrix arrays attached to hosts

The `symcfg list` command displays the Symmetrix arrays attached to your hosts and provides SRDF information, such as:

- SRDF devices on the Symmetrix arrays
- SRDF device states
- Number of invalid tracks for the source and the target devices

To view the Symmetrix arrays attached to your host, type:

```
symcfg list
```

Viewing command results

After executing an SRDF control operation, use the `symrdf query` command to check the command impact on the selected devices, device groups, and composite groups.

For example, to view the command results on the `prod` device group, type:

```
symrdf query -g prod
```

Viewing time estimate for pair synchronization

Both the `symrdf query` and `symrdf verify` have a `-summary` option that provides a condensed display showing the pair state, number of invalid tracks on the source and target, the synchronization rate, and the estimated time remaining for SRDF pair synchronization.

For example, to verify the time remaining for synchronization of the `prod` device group, type:

```
symrdf verify -g prod -summary -i 60 -c 2
```

Showing SRDF device settings

The `symdev show` command displays information about SRDF devices, such as:

- SRDF device type and its group number
- Whether it is paired with a diskless, concurrent, or cascaded device
- Whether it has a standard/thin relationship
- If the R2 device is larger than its R1
- Whether SRDF/A group-level and/or device-level write pacing is currently activated and supported for the SRDF/A session
- Whether it is pace-capable

For example, to show settings about device 11a0 on Symmetrix array 341, type:

```
symdev show 11a0 -sid 341
```

Listing SRDF group settings

The `symcfg list` command with the `-rdfg` option provides information about SRDF group-level settings for a specific group or all groups on a Symmetrix array, such as:

- Group type
- Director configuration
- Group flags, including auto link recovery, link domino, SRDF/Star mode, SRDF software and hardware compression, and SRDF single round trip
- SRDF flags, including consistency and SRDF status and mode

For example, to list the settings for all SRDF groups on Symmetrix 341, type:

```
symcfg list -sid 341 -rdfg all
```

Listing directors with a specific group number

The `symcfg list` command can list directors defined with a specific SRDF group number.

For example, to list all the directors with SRDF group number 54 on Symmetrix 341, type:

```
symcfg list -ra all -rdfg 54 -sid 341
```

Listing composite groups

The `symcg list` command lists the composite groups by name, and by group type. To view all composite groups created in your host database file, type:

```
symcg list
```

If the type of group is RDF1, RDF2, or RDF21, the group is an SRDF composite group. A composite group may contain one or more SRDF groups.

Listing SRDF/Star composite groups

The `symstar list` command displays all the SRDF/Star composite groups visible to the host. By using the `-local` option with this command, only the locally-defined SRDF/Star composite groups are listed.

For example, to list the composite groups local to your host, type:

```
symstar list -local
```

For information on SRDF/Star configurations, refer to the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*.

Listing SRDF devices

The `symrdf list` command and its options display SRDF devices, as described in Table 9.

Table 9 Command options for symrdf list (page 1 of 2)

Option	Definition
-all	Lists all mirrors of the selected SRDF devices.
-bcv	Lists only BCV devices.
-c	Specifies the number (count) of times to repeat the operation, displaying results appropriate to the operation at each iteration.
-cascade	Lists all R21 devices and the R1 and R2 devices with which they are paired. Using the -cascade with either the `-R1`, `-R2`, or `-R21` limits the display to only R1, R2, or R21 devices that are participating in a cascaded SRDF relationship. For example, to list all the devices in a cascaded SRDF configuration in SRDF group 1 on Symmetrix array 333, type: symrdf -sid 333 -rdfg 1 -cascade list To filter the list to only include the R2 devices in a cascaded SRDF configuration in SRDF group 1 on Symmetrix array 333, type: symrdf -sid 333 -rdfg 1 -cascade -R2 list Because R21 devices and the devices with which they are paired are considered concurrent SRDF devices, using the -concurrent flag will display these devices.
-concurrent	Lists concurrent SRDF (RDF11, RDF22, and RDF21) devices as well as SRDF devices paired with a concurrent SRDF device. When used with -R1, lists RDF11 devices and RDF1 devices that are paired with a concurrent SRDF device. When used with -R2, lists RDF22 devices and RDF2 devices that are paired with a concurrent device.
-consistency	Lists the SRDF consistency state when listing SRDF devices. Beginning with Enginuity 5773.150, you can issue this option from both the R1 and R2 sides. For example, to show the consistency state in the list of all the SRDF devices in Symmetrix array 333, type: symrdf -sid 333 -consistency list
-cons_exempt	Lists devices that are consistency exempt or are paired with devices that are consistency exempt.
-dir	Lists the local Symmetrix directors (separated by commas), such as, 1a, 1b, and so on.
-diskless_rdf	Lists diskless SRDF devices as well as devices paired with diskless SRDF devices. When used with -R1, lists RDF1 devices that are either diskless or that are paired with a diskless device. When used with -R2, lists RDF2 devices that are either diskless or are paired with a diskless device. When used with -R21, lists RDF21 devices that are either diskless or that are paired with a diskless device.
-dup_pair	Lists SRDF devices that are paired with the same SRDF type. For example, to list all of the duplicate pair devices in Symmetrix array 333, type: symrdf -sid 333 -dup_pair list Existing duplicate pair devices could result from an SRDF/Star failover scenario or a configuration change.
-dynamic	Lists devices configured as dynamic SRDF. This option allows additional qualifiers of `-R1`, `-R2`, or `BOTH`.

Table 9 Command options for symrdf list (page 2 of 2)

Option	Definition
-half_pair	Lists devices whose partner is not an SRDF device. For example, to list all of the half pair devices in Symmetrix array 333, type: symrdf -sid 333 -halfpair list Existing half pair devices could result from an SRDF/Star failover scenario, a `half_deletepair` operation, or a configuration change.
-nobcv	Lists standard SRDF devices only (excludes SRDF BCV devices).
-R1 -R2 -R21	Lists devices of RDF1 types (`-R1`), RDF2 types (`-R2`), or RDF21 types (`-R21`), respectively.
-rdfa	Lists devices that are SRDF/A-capable.
-rdfa_not_pace_capable	Lists devices participating in the SRDF/A session that are not pace-capable
-rdfa_wpace_exempt	Lists devices that are exempt from group-level write pacing
-rdfg	Lists all devices within a specified SRDF group.
-resv	Lists SRDF devices with SCSI reservations. For example, to list all the SRDF devices in Symmetrix array 333 that have SCSI reservations, type: symrdf -sid 333 -resv list
-star_mode	Lists device that are SRDF/Star protected. For more information, refer to the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*.

CHAPTER 3
Dynamic SRDF Operations

This chapter describes how to create and manage dynamic groups and device pairs using the SYMCLI. Topics include:

- Dynamic SRDF group operations 92
- Dynamic SRDF device pair operations 99

Dynamic SRDF group operations

SRDF groups provide a collective data transfer path linking devices of two separate Symmetrix arrays. These communication and transfer paths are used to synchronize data between the R1 and R2 device pairs associated with the SRDF group. At least one physical connection must exist between the two Symmetrix arrays within the fabric topology, as shown in Figure 12.

SRDF groups can be created on demand while the Symmetrix array is in operation. Previously, *static* SRDF groups could only be defined at the time of the array configuration. You can add, modify, and delete dynamic SRDF groups (RA groups) in a switched fabric SRDF environment.

The maximum number of SRDF groups and SRDF groups per SRDF director varies by Enginuity version:

- Enginuity 5772 and 5773, supports a maximum of 128 SRDF groups, and up to 32 SRDF groups per SRDF director.
- Enginuity 5874 and higher supports a maximum of 250 SRDF groups and up to 64 SRDF groups per SRDF director.

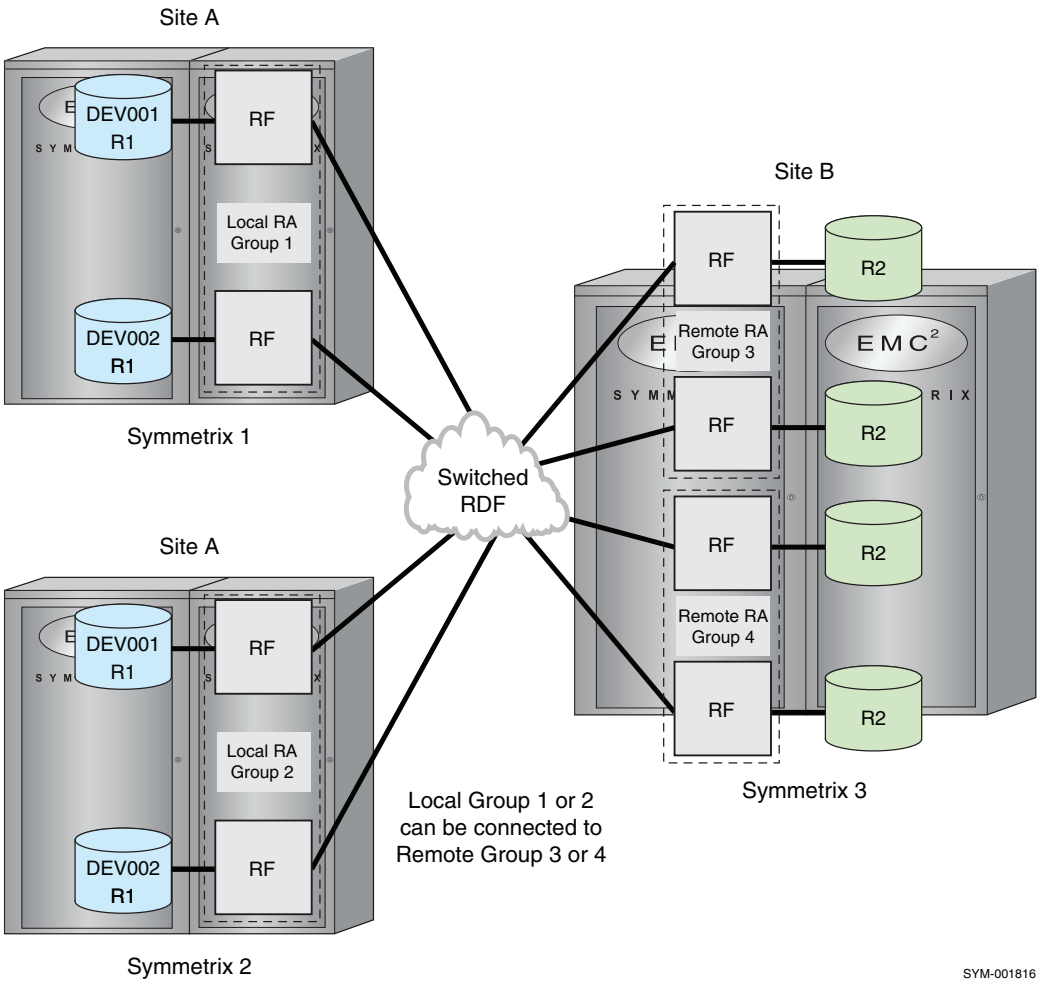


Figure 12 SRDF group topology in a switched SRDF solution

Physical point-to-point fibre connections, shown in Figure 13, are **not** currently supported, even if the SRDF connections are configured as "switched."

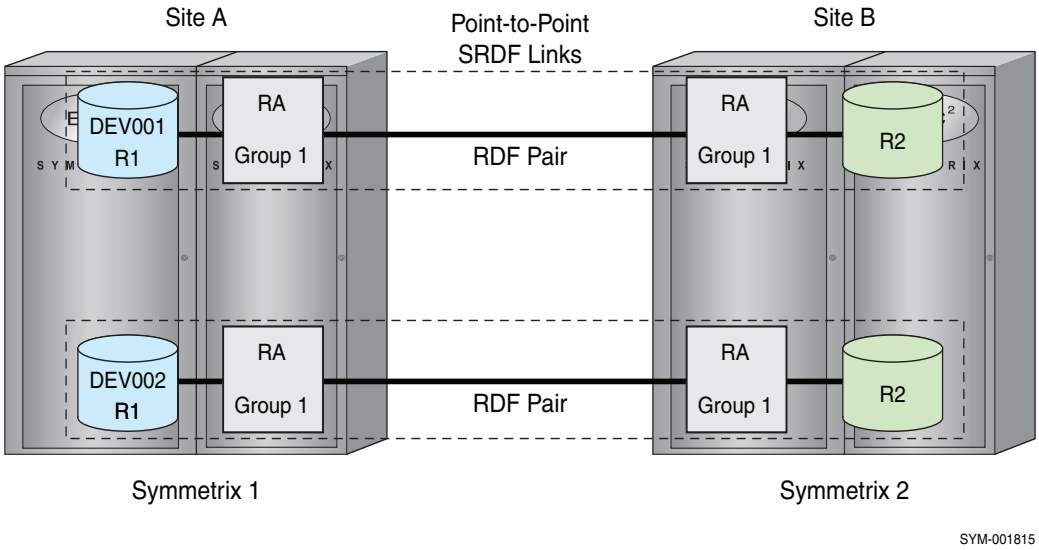


Figure 13 SRDF group topology in a point-to-point SRDF solution

Dynamic SRDF group capability provides flexibility within your SRDF environment to change multiple remote mirroring connections for dynamic devices.

Note: Dynamic SRDF devices cannot currently be used with Parity RAID (RAID/S).

Adding dynamic groups

The `symrdf addgrp` command creates a dynamic SRDF group that represents an additional SRDF link between two Symmetrix arrays.

The following are requirements for adding a dynamic SRDF group:

- The `dynamic_rdf` parameter must be set in your Symmetrix configuration. See the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide* for information on how to set Symmetrix metrics.
- The local or remote Symmetrix array must not be in the symavoid file. If either one of these arrays is in this file, an error message is generated.

Adding a dynamic SRDF group creates an empty group. Dynamic groups must be added one at a time. Adding multiple dynamic groups can be executed in a script, but must be done one group at a time. Once the group is created, you can then add dynamic SRDF pairs to it. A group label must be specified when adding a dynamic group. Only one dynamic group operation (`addgrp`, `modifygrp`, `removegrp`) can be executed at a time before another can be attempted.

With Enginuity 5876 Q4 2012 SR and higher, you can use the `symsan list -sanrdf` command to view the SRDF topology. You can then use this information to choose director pairs when you create the first dynamic SRDF group between two Symmetrix arrays.

For example, to determine which remote directors are visible from Symmetrix array 6180, type:

```
symsan -sanrdf -sid 6180 -dir all

Symmetrix ID: 000194906180

    Flags                Remote
--- ------- ---------------------------------
    Dir Lnk
Dir CT  S   Symmetrix ID Dir WWN
--- --- --- ------------ --- ----------------
12A SO  C   000192606240 13A C465090872090050
14A SO  C   000192602586 15A C465090872016879


Legend:
  Director:
    (C)onfig : S = Fibre-Switched, H = Fibre-Hub
               G = GIGE, - = N/A
    S(T)atus : O = Online, F = Offline, D = Dead, - = N/A

  Link:
    (S)tatus : C = Connected, P = ConnectInProg
               D = Disconnected, I = Incomplete, - = N/A
```

In this example, the output shows that director 13a on Symmetrix array 6240 is visible from director 12a on Symmetrix array 6180

The following example adds a new dynamic SRDF group (`dyngrp4`), which represents the SRDF links between two Symmetrix arrays (`6180` and `6240`). It adds dynamic SRDF group 4 on the local Symmetrix array (`6180`), and SRDF group 4 on the remote Symmetrix array (`6240`). The specified group label (`dyngrp4`) can later be used to modify or delete the group. Directors are specified for both the local (`12a`) and remote (`13a`) Symmetrix arrays:

```
symrdf addgrp -sid 6180 -rdfg 4 -label dyngrp4 -dir 12a
-remote_rdfg 4 -remote_sid 6240 -remote_dir 13a
```

When creating dynamic SRDF groups between two Symmetrix arrays, it is important to understand the network topology when choosing director endpoints. If using fibre protocol, the director endpoints chosen must be able to see each other through the Fibre Channel fabric in order to create the dynamic SRDF links. Ensure that the physical connections between the local RA and remote RA are valid and operational. Static and dynamic groups may co-exist on the same RA directors.

After the group is created, dynamic SRDF pairs can be added to it. The following example adds the dynamic SRDF pairs listed in the device file (`dynpairsfile`) to the new dynamic SRDF group 4:

```
symrdf createpair -sid 6180 -rdfg 4 -file dynpairsfile -type R1
-invalidate R2
```

For more information on adding dynamic SRDF pairs, refer to "Dynamic SRDF device pair operations" on page 99.

You can use the `symcfg list -ra all -switched` command to display all SRDF groups on the local Symmetrix array and its remotely connected Symmetrix arrays. SRDF groups are listed as static or dynamic under the group type.

Modifying dynamic groups

Using the `symrdf modifygrp` command, you can modify an existing dynamic SRDF group. The modify command can be used to add or remove supporting directors to or from a dynamic SRDF group. Reassigning directors for SRDF dynamic groups requires that you understand the network fabric topology when choosing director endpoints. The modify command cannot be used to modify existing static groups. You must specify the group label or group number.

The following example modifies a dynamic group (`dyngrp4`) to remove a supporting director 13a assigned from the group on the local Symmetrix array 6180:

```
symrdf modifygrp -sid 6180 -label dyngrp4 -remove -dir 13a
```

The following example modifies a dynamic group (`dyngrp4`) to add (assign) a supporting director 12a to the group on the local Symmetrix 6180:

```
symrdf modifygrp -sid 6180 -label dyngrp4 -add -dir 12a
```

Note: When adding a director to a dynamic group, the specified director for the local Symmetrix array must be online and a physical link to one online director in the remote Symmetrix array must exist.

IMPORTANT

Making physical cable changes within the SRDF environment may disable the ability to modify and delete dynamic group configurations.

Renaming the label of a dynamic group

You can rename the label of a dynamic group by using the `symrdf set label` command. This label can be one to 10 characters long.

The following are requirements for renaming a dynamic group label:

- The target group must be a dynamic SRDF group.
- The new label must be unique; it cannot be assigned to existing dynamic groups on either the local or remote Symmetrix array.
- The dynamic group being modified cannot be in the Partitioned state on all director pairs on which it is defined.
- Both the local and remote Symmetrix arrays must be running Enginuity 5876 and higher.

The following resets the label of the SRDF group 4 on Symmetrix 6180 to `newdyngrp4`:

```
symrdf -sid 6180 -rdfg 4 set label newdyngrp4
```

Removing dynamic groups

To remove a dynamic group, both sides of the SRDF configuration must be defined and reachable, and you must empty the group of all assigned devices using the `symrdf deletepair` command. At least one physical connection between Symmetrix arrays must exist. Deleting the dynamic group removes all local and remote director support.

The following example deletes from the group the SRDF dynamic pairs defined in a device file, and then removes the local and remote dynamic SRDF groups created and modified in the previous examples:

```
symrdf deletepair -sid 80 -rdfg 4 -file dynpairsfile

symrdf removegrp -sid 80 -label dyngrp4
```

IMPORTANT

Making physical cable changes within the SRDF environment may disable the ability to modify and delete dynamic group configurations.

Removing a dynamic group from one side of an SRDF configuration

By using the `-symforce` option with `symrdf removegrp`, you can remove an SRDF group from one side of an SRDF configuration as long as the other side is not defined or reachable and the group is empty. If the other side of the SRDF configuration is accessible, you cannot execute this command.

The following example removes `dyngrp4` from Symmetrix 180 on the local side:

```
symrdf removegrp -sid 180 -label dyngrp4 -symforce
```

This functionality requires Symmetrix arrays running Enginuity 5876 and higher.

Dynamically changing Link Domino and Autolink Recovery modes

Link Domino and Autolink Recovery modes are attributes that you set for SRDF groups. In this mode, if SRDF cannot replicate data, the R1 devices are made NR (not ready) to their host, rejecting any I/Os from the host application. This ensures that the data on both the source and remote sites are always in sync.

Autolink Recovery mode is also an SRDF group-level attribute. This mode ensures devices that were ready to their host before a link failure are automatically restored to a ready state once the link is up and running.

By default, Link Domino and Autolink Recovery modes are enabled or disabled at the local side of an SRDF group. You can use the `both_sides` option to enable and disable these modes on both the local and remote sides of an SRDF group. To set these attributes, the remote side must be reachable.

For example, to set the Link Domino mode on both sides of group 4 on Symmetrix array 6180, type:

```
symrdf -sid 6180 -rdfg 4 set rdfg -domino on -both_sides
```

For example, to set the Autolink Recovery mode on both sides of group 4 on Symmetrix array 6180, type:

```
symrdf -sid 6180 -rdfg 4 set rdfg -autolink_recovery on
-both_sides
```

Specifying a link limbo for a dynamic group

Link limbo is an advanced user feature that allows you to set a specific length of time for Enginuity to wait when a down link is detected before updating the link status.

You can specify a link limbo value on the local side or both the local and remote sides of a dynamic SRDF group. If the link status is still not ready after the link limbo time expires, devices are marked not ready to the link.

The link limbo value can be set from 0 to 120 seconds.

The default value is 10 seconds.

For example, to set the link limbo value to one minute for both sides of SRDF group 4 on Symmetrix array 6180, type:

```
symrdf -sid 6180 -rdfg 4 set rdfg -limbo 60 -both_sides
```

Note: Because the setting of this value affects the application timeout period, it is not recommended to set while running in synchronous mode.

To protect from session drops after the maximum link limbo time, you can choose to enable the Transmit Idle feature. For more information on Transmit Idle, refer to "Using transmit idle" on page 133.

Setting SRDF group attributes

Use the following syntax to set SRDF group attributes:

```
symrdf -sid <SymmID> -rdfg <GrpNum> [-v] [-symforce]
   [-noprompt] [-i <Interval>] [-c <Count>]

.............

set rdfg
   <[-limbo <LinkLimboValue>]
   [-domino <State>]
   [-autolink_recovery <State>]
   [-hwcomp <State>]
   [-swcomp <State>]>
   [-both_sides]
```

To set these group attributes, the remote side must be reachable.

Where:

- `-limbo` (`-lim`) is the length of time to wait before updating the SRDF link status as down. This time begins when Enginuity detects the SRDF link is down. If the link status is still not ready when this time expires, devices are marked as not ready to the link. Valid values for the link limbo are zero through 120 seconds with a default of 10 seconds. This option is for advanced users only.
- `-domino` (`-dom`) ensures data on the source and target devices of a group are always in sync. Valid state values are on and off (default).
- `-hwcomp` (`-hwc`) enables or disables hardware compression for a group. Valid state values are on and off (default).
- `-swcomp` (`-swc`) enables or disables software compression for a group. Valid state values are on and off (default).
- `-autolink_recovery` (`-aut`) enables or disables autolink recovery on the source side of an SRDF/A group unless `-both_sides` is used. Valid state values are on and off (default).
- `-both_sides` applies the group attributes to both the source and target sides of an SRDF session. Otherwise, these attributes are only applied to the source side.

To set limbo to thirty seconds and turn off Link Domino and Autolink Recovery modes for SRDF group 12, type:

```
symrdf -sid 134 -rdfg 12 set rdfg -limbo 30 -domino off -autolink_recovery off

An RDF Set 'Attributes' operation execution is in progress for RDF group 12.
Please wait...

    SRDF/A Set Link Limbo (1134,012)..............................Started.
    SRDF/A Set Link Limbo (1134,012)..............................Done.
    SRDF/A Set Link Domino (1134,012)............................Started.
    SRDF/A Set Link Domino (1134,012)............................Done.
    SRDF/A Set Auto Link Recovery (1134,012).....................Started.
    SRDF/A Set Auto Link Recovery (1134,012).....................Done.
The RDF Set 'Attributes' operation successfully executed for RDF group 12.
```

To turn on software compression and turn off hardware compression on both sides of the SRDF group 12, type:

```
symrdf -sid 134 -rdfg 12 set rdfg -swc on -hwc off -both_sides

An RDF Set 'Attributes' operation execution is in
progress for RDF group 12. Please wait...

    SRDF/A Set Software Compression (1134,012)...................Started.
    SRDF/A Set Software Compression (1134,012)...................Done.
    SRDF/A Set Software Compression (0601,015)...................Started.
    SRDF/A Set Software Compression (0601,015)...................Done.
    SRDF/A Set Hardware Compression (1134,012)...................Started.
    SRDF/A Set Hardware Compression (1134,012)...................Done.
    SRDF/A Set Hardware Compression (0601,015)...................Started.
    SRDF/A Set Hardware Compression (0601,015)...................Done.
The RDF Set 'Attributes' operation successfully executed
for RDF group 12.
```

You can also use the `symconfigure` command to set SRDF group attributes. For more information, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Dynamic SRDF device pair operations

SRDF device pairing was previously limited to the static SRDF pairs set at Symmetrix configuration time. Dynamic SRDF enables the creation and deletion of SRDF pairs while the Symmetrix array is in operation. Once established, the new SRDF pairs can be synchronized and managed in the same way as configured SRDF pairs.

Creating a device file

After identifying the dynamic SRDF devices, create a device file containing two columns for the pairing of the R1 and R2 devices (*SymDevnames)*. List each SRDF pair on a separate line in the device file. The `-type` option of the `symrdf createpair -file` command indicates whether the devices in the first column are R1 or R2.

The following example uses the vi text editor to create the `RDFG148` device file consisting of dynamic SRDF pairs for the local and remote Symmetrix arrays. The R1 devices are in the first column with their corresponding remote R2 devices in the second column. For example, the first line shows the pairing of 0060 (R1) with 0060 (R2):

```
vi RDFG148
0060 0060
0061 0061
0062 0062
0063 0063
0064 0064
0065 0065
0066 0066
```

Creating dynamic SRDF device pairs

This section explains the various ways to create dynamic SRDF device pairs. Table 10 lists the valid device type combinations to use when creating an SRDF pair.

Table 10 Device type combinations for creating SRDF pairs

Device 1	Device 2
Standard	Standard
Thin	Thin
Standard	Diskless
Thin[1]	Diskless[1]
Thin[2]	Standard[3]

1. FBA devices must be on a Symmetrix array running Enginuity 5875 or higher. CKD devices are not supported.
2. FBA devices must be on a Symmetrix array running Enginuity 5875 or higher. CKD devices must be on a Symmetrix array running Enginuity 5876 Q42012 SR or higher.
3. Must be on a Symmetrix array running Enginuity versions 5671, 5773.50154, or 5875 and higher.

Default SRDF mode

For pre-Solutions Enabler V7.4, the default SRDF mode is synchronous when you create device pairs without setting a mode.

For Solutions Enabler V7.4 and higher, the default mode is adaptive copy disk when you create device pairs without setting a mode. You can change this default mode to synchronous by resetting the SYMAPI_DEFAULT_RDF_MODE parameter in the options file.

Note: Regardless of the SYMAPI_DEFAULT_RDF_MODE setting, diskless R21->R2 device pairs are always adaptive copy write pending. This type of pairing only accepts adaptive copy write pending or asynchronous.

For descriptions of SRDF modes, see "Understanding and setting SRDF modes of operation" on page 32.

To create dynamic device pairs, use the following syntax:

```
symrdf -file Filename -sid SymmID -rdfg GrpNum [-bypass] [-noprompt]
   [-i Interval] [-c Count]
   [-v|-noecho] [-force] [-symforce] [-star]

createpair -type <R1|R2>
   <<-invalidate <R1|R2> | -establish | -restore> [-nowd] |
   [-rp] -format [-establish]>
   [-rdf_mode <sync|semi|acp_wp|acp_disk|async>]
   [-g NewDg] [-remote] [-cons_exempt]
```

Where:

- The invalidate option (`-invalidate [R1|R2]`) marks the R1 devices or R2 devices in the list to be the invalidated target for a full device copy once the SRDF pairs are created.

- The establish option (`-establish`) begins copying data to invalidated targets, synchronizing the dynamic SRDF pairs in the device file once the SRDF pairs are created.
- The restore option (`-restore`) begins copying data to the source devices, synchronizing the dynamic SRDF pairs in the device file once the SRDF pairs are created.
- The nowd option (`-nowd`) bypasses the check explained in "Verifying host cannot write to target devices" on page 101.
- The rp option (`-rp`) allows the operation even when one or more devices are tagged for RecoverPoint.
- The format option (`-format`) clears all tracks on the R1 and R2 sides to ensure no data exists on either side, and makes the R1 read write to the host. You can specify this option with `-establish`, `-type`, `-rdf_mode`, `-cons_exempt`, and `-g`. When used with `-establish`, the devices become read write on the SRDF link and are synchronized.
- The rdf option (`-rdf_mode)` sets the SRDF mode of the pairs to be one of the following: synchronous (sync), semi-synchronous (semi), asynchronous (async), adaptive copy disk mode (acp_disk), or adaptive copy write Pending mode (acp_wp). Adaptive Copy Disk is the default mode unless overridden by the SYMAPI_DEFAULT_RDF_MODE options file setting.
- The g option (`-g`) specifies the device group name to be created with the devices in the device file.
- The remote option (`-remote`) requests a remote data copy flag with failback, failover, restore, update, and resume.
- The consistency exempt (`-cons_exempt`) option allows devices to be added without affecting the state of the SRDF/A session or requiring that other devices in the session be suspended.

Verifying host cannot write to target devices

The options file contains the SYMAPI_RDF_CHECK_R2_NOT_WRITABLE parameter that checks if the host cannot write to the R2 devices during `createpair` operations (other than `createpair -invalidate <R1|R2>`). This parameter is disabled by default.

The `symrdf createpair` command has the `-nowd` option to bypass this check. It applies to the R2 devices for all `createpair` actions and to the R1 devices for the `createpair -invalidate R1`. It also applies to the source (R1) devices for `-invalidate R1` and to the target (R2) devices for `-invalidate R2`.

Creating dynamic pairs using a device file

The following example dynamically creates SRDF device pairs from `devicefile`, ignores the check to see if the host can write to its targets, and by not specifying the SRDF mode, uses the default mode (adaptive copy disk):

```
symrdf createpair -sid 123 -file devicefile -type r1 -rdfg 10 -nowd
```

Creating dynamic pairs with establish

The following `createpair -establish` command creates the dynamic device pairs and begins copying data to its targets, synchronizing the device pairs listed in the device file.

```
symrdf createpair -file devicefile -sid 55 -rdfg 1 -type R1
-establish
```

For the `createpair -establish` option, the R2 may be set to read/write disabled (not ready) if `SYMAPI_RDF_RW_DISABLE_R2=ENABLE` is set in the options file. For more information, refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

Creating dynamic pairs with format

The format option (`-format`) clears all tracks on the R1 and R2 sides to ensure no data exists on either side, and makes the R1 read write to the host. When you use this option to create dynamic pairs, an application cannot write to these devices.

The `symrdf createpair -format` option has the following restrictions:

- The Symmetrix arrays must be running Enginuity 5876 and higher.
- Not supported in concurrent and cascaded SRDF environments.
- The R1 and R2 cannot be mapped to a host.

To create dynamic pairs using `-format`, type:

```
symrdf createpair -sid 66 -format -file devicefile -type r1 -rdfg 117
-rdf_mode sync -nop
```

Creating dynamic pairs with invalidate

The following example dynamically creates a number of SRDF paired devices and invalidates the targets:

```
symrdf createpair -sid 55 -file devicefile -rdfg 1 -type R1
-invalidate r2 -nowd
```

The above `symrdf createpair` command creates new SRDF pairs from the list of devices in the device file. For this example, a file called `devicefile` identifies the first-column devices that are in Symmetrix 55 as R1 type devices. The R2 devices are invalidated with this create pair example and the SRDF pairs become members of SRDF group 1. Upon execution of this command, this pairing information is added to the SYMAPI database file on the host. The validation exempt check (`-nowd`) option allows you to bypass the validation check to ensure that the target of operation is write disabled to its host.

You can now establish the SRDF pairs in the list, which copies data to the invalidated target devices. You can then query to check the progress of the establish operation. Once synchronized, you can perform various SRDF operations on SRDF pairs listed in the device file:

```
symrdf -sid 55 -file devicefile establish -rdfg 1
symrdf -sid 55 -file devicefile query -rdfg 1
```

Creating dynamic pairs for a restore

You can perform a restore operation to copy data back to the R1 source devices by including the `-restore` option in the `createpair` command line as follows:

```
symrdf createpair -sid 55 -file devicefile -rdfg 1 -type R1 -restore
```

Once the SRDF device pairs are created, the restore operation begins copying data to the source devices, synchronizing the dynamic SRDF device pairs listed in the device file.

This `createpair` operation is rejected when:

- The device is in one of the following BCV pair states: Synchronized, SyncInProg, Restored, RestoreInProg, SplitInProg.
- The device is the source or target of a TimeFinder/Snap operation.
- There is a background BCV split operation in progress.
- Devices are in the backend not ready state.
- There is an optimizer swap in progress on a device.
- The emulation type is not the same (such as, AS/400 has specific pairing rules).
- There are existing local invalid tracks on either the local or remote device.
- The SRDF/A session is active and `-cons_exempt` is not specified.
- The SRDF group is in asynchronous mode and the devices being added are not the same SRDF type R1 or R2.
- The SRDF group is in asynchronous mode with the SRDF links suspended and the `-establish` or `-restore` option is selected.
- The SRDF group is enabled for SRDF consistency protection.
- The operation involves one or more of the following unsupported devices: VCM DB, SFS, RAD, DRV, RAID-S, WORM-enabled devices, 4-way mirror, Meta member, PPRC mainframe system.

Creating dynamic concurrent SRDF pairs

You can dynamically create concurrent SRDF pairs using the `symrdf createpair` command. This feature allows a second remote mirror to be dynamically added by converting a dynamic R1 device to a concurrent SRDF device. Two remote mirrors are supported for any dynamic R1 device.

With Enginuity 5875 or higher, both mirrors of a concurrent R1 device can be operating in SRDF/A mode.

For information about a concurrent SRDF configuration, refer to "Concurrent SRDF Operations" on page 141.

The following rules apply when creating a dynamic concurrent SRDF pair:

- The SRDF BCVs designated as dynamic SRDF devices are not supported.
- The two SRDF mirrors of the concurrent device must be assigned to different SRDF groups.
- The concurrent dynamic SRDF, dynamic SRDF, and concurrent SRDF states must be enabled in your Symmetrix configuration.
- With the `-restore` selection, the `-remote` option is required if the link status for the first created remote mirror is read/write.

To dynamically create a second remote mirror using the `symrdf createpair` command, you must create two separate device files: one file containing the first set of R1/R2 device pairs, and a second device file listing the same R1 device paired with a different remote R2 device.

To create a concurrent mirror for the `createpair` example previously described, you can use the following example:

```
symrdf createpair -sid 55 -rdfg 2 -file devicefile2 -type R1
-invalidate R1|R2
```

In the previous example, the `symrdf createpair` command creates new SRDF pairs from the list of devices in a second device file. For this example, a file called `devicefile2` identifies the first-column devices that are in Symmetrix 55 as R1 type devices. The second-column devices become the second remote mirror devices. The SRDF device pairs will become members of SRDF group 2. Also, you can execute a restore/remote operation to restore the standard R2 type devices:

```
symrdf createpair -file devicefile2 -sid 55 -type R1 -restore -remote
```

Note: The concurrent mirror device pairs must belong to a separate RA group than those defined in the first device file pairing.

Blocking createpair when R2 is larger than R1

When an R2 device is larger than its corresponding R1 device, you cannot use this R2 device to restore or failover to the R1. To avoid creating such an SRDF pair, Solutions Enabler provides an option called `SYMAPI_RDF_CREATEPAIR_LARGER_R2` in the options file that you can `ENABLE` (the default) or `DISABLE`. If set to `DISABLE` when attempting to create a pair with an R2 larger than its R1, Solutions Enabler blocks the `createpair` operation.

Deleting dynamic SRDF pairs

The `deletepair` command cancels the dynamic SRDF pairs, removes the pairing information from the Symmetrix array and the SYMAPI database, and changes these devices to non-SRDF devices (except for concurrent SRDF pairs).

You must suspend the SRDF links using the `symrdf suspend` command before performing the `symrdf deletepair` command.

The following commands suspend the SRDF links for the devices listed in `devicefile` on Symmetrix 000125600055, and then deletes these SRDF pairs from `devicefile`. The `-rdfg 2` specifies the SRDF group number by which the pairs communicate:

```
symrdf suspend -sid 55 -file devicefile -rdfg 2
symrdf deletepair -sid 55 -file devicefile -rdfg 2
```

Note: To prevent a device group or a composite group from becoming invalid, first remove the devices from the group before performing the `deletepair` action on a device file.

After execution of the `symrdf deletepair` command, the dynamic SRDF pairs are canceled.

Note: Solutions Enabler removes the pairing information from the Symmetrix array and the SYMAPI database, and changes the devices to non-SRDF devices (except for concurrent SRDF pairs).

The `deletepair` operation is rejected when:

- The device is in one of the following BCV pair states: Synchronized, SyncInProg, Restored, RestoreInProg, and SplitInProg.
- There is a background BCV split operation in progress.
- Devices in the backend are not in the ready state.
- There is an optimizer swap in progress on a device.
- SRDF consistency protection is enabled and the devices were not suspended with the `-cons_exempt` option.
- The SRDF links are not suspended.

Deleting dynamic pairs and clearing local invalid tracks

When you use `-symforce` with `symrdf deletepair`, the SRDF relationship between the R1 and R2 devices is removed and any local invalid tracks on these devices are cleared. For example:

```
symrdf suspend -sid 55 -rdfg 112 -file devicefile
symrdf deletepair -sid 55 -rdfg 112 -symforce -file devicefile
```

This functionality does not support diskless devices and does not delete any device pairs containing R11, R21, or R22 devices. The devices must reside on Symmetrix arrays running Enginuity 5876 and higher.

Deleting one-half of a pair

The `half_deletepair` command allows you to dynamically remove the SRDF pairing relationship between R1/R2 device pairs. One-half of the specified device pair is converted from an SRDF device to a regular device. The command can be specified using a device file or device group. When specified using a device file, all devices listed in the first column of the file are converted to regular devices (non-SRDF). Table 55 on page 498 lists the applicable SRDF pair states for control operations.

For example, to remove the SRDF pairing from device group `Prod` and convert the devices assigned to `Prod` to regular (non-SRDF) devices, leaving their remote partners as SRDF devices, type:

```
symrdf -g Prod half_deletepair
```

To remove the SRDF pairing of SRDF group 4 on Symmetrix 1123 and convert one-half of those device pairs to regular (non-SRDF) devices, type:

```
symrdf half_deletepair -sid 123 -rdfg 4 -file devicefile
```

You can use the `symrdf list -half_pair` command to list all half pair devices for a specified Symmetrix or SRDF group. Existing half pairs can also be the result of a previous `symrdf failover` operation or a configuration change.

The `symrdf half_deletepair` is rejected when:

- The device is in one of the following BCV pair states: Synchronized, SyncInProg, Restored, RestoreInProg, and SplitInProg.
- There is a background BCV split operation in progress.
- Devices in the backend are not in the ready state.
- There is an optimizer swap in progress on a device.
- SRDF consistency protection is enabled and the devices were not suspended with the `-cons_exempt` option.
- The SRDF links are not suspended.

Displaying dynamic SRDF devices

To display dynamic SRDF devices, use the `symdev list` command with the `-dynamic` option as follows:

```
symdev list -dynamic [-R1] [-R2] [R21]
```

From the displayed list, determine which dynamic devices on the source Symmetrix array you would like to pair with dynamic devices on the target Symmetrix array. If `-R1`, `-R2` or `R21` is not specified, all devices that are SRDF-capable are displayed.

Grouping dynamic pairs with a device file

By grouping the devices in a device file, you can create dynamic pairs on which you can perform control operations. To implement this functionality, do the following:

1. List your device pairings in a device file, and then create dynamic SRDF pairs, applying the `-g GroupName` option to the command line. This adds the devices listed in the device file to a device group (*NewGrp*) as follows:

   ```
   symrdf createpair -sid 55 -rdfg 2 -file devicefile -type rdf1
   -invalidate r2 -g NewGrp
   ```

2. Perform SRDF control operations on the dynamic SRDF pairs within the device group. For example, establish the group:

   ```
   symrdf -g NewGrp establish
   ```

The `symrdf createpair` command created dynamic SRDF pairs from the device file and added the pairs to a device group called NewGrp. The `symrdf establish` command then performed an establish operation on the dynamic SRDF pairs in the device group `NewGrp`. All SRDF commands for these dynamic pairs can now be executed within the context of the `NewGrp` device group.

Deleting SRDF dynamic pairs

The `symrdf deletepair` command changes the devices within a group to non-SRDF devices, and changes the SYMAPI device group to a regular device group (except when an SRDF concurrent pair exists).

For example, to change the SRDF devices in `NewGrp` to non-SRDF devices:

```
symrdf deletepair -g NewGrp
```

Note: The SRDF links must be suspended prior to deleting SRDF pairs using this functionality.

If additional devices were added to the device group before the `symrdf deletepair` command is issued, those added devices are also changed to non-SRDF devices, and the device group is changed to a regular device group, only if the added devices contained within it were dynamic devices. If the device group contained both SRDF and non-SRDF devices, the device group would be changed to an Invalid state.

Moving dynamic SRDF pairs

Beginning with Enginuity 5773, you can move dynamic SRDF devices from one SRDF group to another. Enginuity versions prior to 5773 require that you delete the original SRDF pair, create the new pair, and then move the SRDF pair to a new group. The devices in the new group then required a full synchronization.

This functionality allows the devices to change groups without deleting the SRDF pair and recreating it. There is also no need to fully resynchronize the devices when performing the move because the current invalid track counters on both R1 and R2 stay intact. Table 55 on page 498 lists the applicable SRDF pair states for control operations.

SRDF pairs can be moved using a device file or device group. The new SRDF group option (`-new_rdfg`) is always required. The consistency exempt (`-cons_exempt`) option allows devices to be moved into an active SRDF/A session without affecting the state of the session or requiring that other devices in the session be suspended.
A prior suspension of devices using the `-cons_exempt` option is required in order to move devices out of an active SRDF/A session without affecting the state of the session. After a successful move, the pair state is unchanged.

The `movepair` operation has the following restrictions:

- A device cannot move when it is enabled for SRDF consistency.
- A device cannot move if it is in asynchronous mode when an SRDF/A cleanup or restore process is running.
- When moving one mirror of a concurrent R1 or an R21 device to a new SRDF group, the destination SRDF group must not be the same as the one supporting the other SRDF mirror.
- When issuing a full `movepair` operation, the destination SRDF group must connect the same two Symmetrix arrays as the original SRDF group.
- If the destination SRDF group is in asynchronous mode, the SRDF group type of the source and destination group must match. In other words, in asynchronous mode, devices can only be moved from R1 to R1, or from R2 to R2.
- If the destination SRDF group is supporting an active SRDF/A session, the `-cons_exempt` option must be specified.
- If the original SRDF group is supporting an active SRDF/A session, the device pairs being moved must have been suspended using the `-cons_exempt` option.

Moving one-half of an SRDF pair

Moving only one side of an SRDF pair from one SRDF group to another is also supported. This is called a half move pair (`half_movepair`). Table 55 on page 498 lists the applicable SRDF pair states for control operations.

After a successful `half_movepair` the pair state can go from partitioned to a different state or vice versa. For example, when a `half_movepair` action results in a normal SRDF pair configuration, the resulting SRDF pair state will be Split, Suspended, FailedOver or Partitioned.

Cascaded SRDF device moves

Devices that are in a cascaded SRDF relationship can be moved independent of device pair state in the other SRDF link (one not being moved). However, the `-force` flag is required if the other SRDF link (the one not being controlled) is enabled for SRDF consistency protection.

With a device pair that is participating in a cascaded SRDF relationship, the pair state of the other half of the cascaded relationship may affect whether an SRDF control operation is allowed. Refer to Table 56 on page 502 for the list of control operations and applicable pair states for a cascaded SRDF configuration.

SRDF mode after a movepair

After a `movepair` or `half_movepair` action, the resulting SRDF mode for the moved device is as follows:

- When moving a device *to* an SRDF group that is currently in asynchronous mode, the resulting SRDF mode for the device being moved is asynchronous.
- When moving a device *from* an SRDF group that is in asynchronous mode to an SRDF group that is not in asynchronous mode, the resulting SRDF mode for the device being moved will be adaptive copy disk.
- For cascaded SRDF:
  - If moving the R1-› R21 pair, the resulting SRDF mode is synchronous.
  - If moving the R21-›R2 pair and the R21 is a diskless device, the resulting SRDF mode is adaptive copy write pending. Otherwise, the SRDF mode is adaptive copy disk.

Issuing a dynamic R1/R2 swap

The dynamic R1/R2 swap feature swaps the SRDF personality of the SRDF device designations of a specified device or composite group. This feature can also be performed on devices in SRDF/A mode. Dynamic SRDF must be enabled to perform this operation.

Note: Do not perform a dynamic swap on SRDF/A devices enabled for consistency protection or if the SRDF/A session is actively copying.

With a dynamic swap, source R1 devices become target R2 devices and target R2 devices become source R1 devices.

Dynamic SRDF devices are configured as one of three types: RDF1 capable, RDF2 capable, or both. Dynamic R1/R2 swap capability requires that the devices be configured as both to initiate a swap.

Note: Dynamic swap is not supported on systems where the R2 device is larger than the R1 device.

Issuing a dynamic half-swap

You can issue a `half_swap` command that swaps one half of an SRDF relationship. This command changes an R1 mirror to an R2 mirror or an R2 mirror to an R1 mirror.

The `half_swap operation` has the following restrictions:

- The R2 device cannot be larger than the R1 device.
- A swap cannot be performed during an active SRDF/A session or when cleanup or restore is running.

Swapping cascaded SRDF devices

Swapping of an R21 device in a cascaded SRDF relationship is allowed as long as the R21 device is converted into a concurrent R1 (R11) device. Also, you can convert a concurrent R1 device into an R21 device.

For example, with R2->R11->R2, you can swap either side of the relationship, as follows:

- Swap R2-> to get R1-> R21->R2
- Swap R11-> R2 to get R2-> R21->R1

Both of the above swaps work.

To illustrate the restriction, here is another example set of devices: R1->R21->R2

The following swap is allowed:

Swap R1->R21 to get R2-> R11-> R2

The following swap is **not** allowed:

Swap R21->R2 to get R1->R22-> R1

Displaying SRDF swap-capable devices

To display SRDF devices that have been configured as dynamic SRDF-capable, you can use the `symrdf list` command with the `-dynamic` option as follows:

```
symrdf list -dynamic [-R1] [-R2] [-both]
```

If no option is specified, all SRDF devices that are SRDF-capable will be displayed. Use the `-R1` option to display all dynamic SRDF-capable devices that are configured as capable of becoming R1. Use the `-R2` option to display all dynamic SRDF-capable devices that are configured as capable of becoming R2.

To display a list of dynamic SRDF-capable devices that are configured as capable of becoming R1 or R2, use the `-both` option as follows:

```
symrdf list -dynamic -both
```

From the displayed list, determine which dynamic devices you want to swap.

Swapping SRDF devices

You can execute the `swap` command for device groups, composite groups and device files. Once devices are dynamically swapped, an `incremental establish` operation is initiated and the devices become immediately available on the link. "Establish (incremental)" on page 61 explains this operation.

To perform an R1/R2 swap, use the following command syntax:

```
symrdf -g DgName [-v | -noecho] [-force] [-symforce]
   [-bypass] [-noprompt] [-i Interval] [-c Count]
   [-hop2 | -bcv [-hop2] | -all | -rbcv | -brbcv]
   [-rdfg GrpNum] [-star]
```

The `-bcv` option lets you target just the BCV devices associated with the SRDF device group for the `swap` action. Use `-all` to target both BCV and standard devices. Use nothing to target just the standard devices.

The `-refresh` option marks the source R1 devices or the target R2 devices to refresh from the remote mirror.

The `-hop2` option can be used with device groups for STDs (`-hop2`) and local BCVs (`-bcv -hop2`).

The following example swaps the R1 designation of the associated BCV RDF1 devices within device group `ProdGrpB`. It also marks to refresh any modified data on the current R1 side of these BCVs from their R2 mirrors:

```
symrdf -g ProdGrpB -bcv swap -refresh R1
```

Refreshing the data status

Swapping the R1/R2 designation of the SRDF devices can impact the state of your stored data, as shown here:

- `-refresh R1` — The R2 device holds the valid copy and the R1 device's invalid tracks will be updated using the R2 data.
- `-refresh R2` — The R1 device holds the valid copy and the R2 device's invalid tracks will be updated using the R1 data.

The refresh action indicates which device does **not** hold a valid copy of the data before the swap operation begins. If you determine that the R1 holds the valid copy, the action of `refresh R2` will obtain a count of the tracks that are different on the R2 and will mark these tracks to refresh from the R1 to the R2 device. The result will be the reverse if you choose to `refresh R1` as the option.

Required states before a swap operation

The current states of the various devices involved in the SRDF swap must be considered *before* executing a swap action. Table 11 lists which states are legal for this operation.

Table 11 SRDF device states before swap operation

SRDF state	Source R2 invalids	Target R2 invalids	State after swap
Suspended with R1 Write Disabled	Refresh R1|R2	Refresh R1|R2	Suspended
R1 Updated	Refresh R1	NA	Suspended
Failed Over	Refresh R1	NA	Suspended

Impact on I/O

When swapping source and target attributes I/O is not allowed to the R1 device, but I/O is allowed to the R2 device.

Disabling SYMAPI behavior parameter

In the options file, behavior parameter `SYMAPI_CTRL_OF_NONVISIBLE_DEVS` must be set to `DiSABLE` to prevent control of devices that are not mapped to the user host. The default is `ENABLE`.

Issuing dynamic failover establish

SRDF dynamic devices can be quickly failed over, swapped, and then re-established all within a single command-line operation. This functionality requires that dynamic devices be both RDF1 and RDF2 capable.

When the `symrdf failover -establish` command is issued, the SRDF devices in the group will perform all of the necessary steps in the following order:

- First, the devices will be failed over, making the R2 devices in the group read/write enabled to their local hosts. "Failover" on page 66 provides a detailed explanation of a failover operation.
- After the failover operation has completed, the SRDF pairs will swap personalities (the R1 devices become R2 devices and the R2 devices become R1 devices). "Issuing a dynamic R1/R2 swap" on page 108 provides a detailed explanation with restrictions that apply when performing a dynamic swap operation.
- Once the devices are dynamically swapped, an incremental establish operation is initiated and the devices become immediately available on the link. "Establish (incremental)" on page 61 explains this operation.

The `symrdf failover -establish` command can be used as a composite operation on dynamic SRDF devices to quickly perform the following single control operations together:

- Fail over from the R1 to the R2 or from the R1 to the R21 in a cascaded SRDF device relationship
- Dynamic SRDF swap
- Incremental establish on the swapped SRDF pairs

The `failover restore` operation has the following restrictions:

- Both the R1 and the R2 devices in the failover must be dynamic SRDF devices.
- The R2 device cannot be larger than its R1 device.
- The swap cannot result in a cascaded R21<-->R21 device pair.
- Cannot execute this command on both mirrors of a concurrent R1 device (composite group operation). This swap would convert the concurrent R1 into a concurrent R2, with a restore on both mirrors of that concurrent R2.

Issuing dynamic failover restore

The `symrdf failover -establish` operation does not support devices operating in asynchronous mode with a read/write link because the R2 data is two cycle switches behind the R1 data, and swapping these devices would result in data loss.

The `symrdf failover -restore` swaps the R1 and R2 and also restores the invalid tracks on the new R2 side (formerly R1) to the new R1 side (formerly R2). You can execute this command for device groups, composite groups and device files. The devices in this failover can be using synchronous or asynchronous links.

The following is the `failover -restore` command syntax for device groups using SRDF/A:

```
symrdf -g DgName [-v | -noecho] [-force] [-symforce]
   [-bypass] [-noprompt] [-i Interval] [-c Count]
   [-hop2 | -bcv [-hop2] | -all | -rbcv | -brbcv]
   [-rdfg GrpNum] [-star]

failover [- immediate | -establish | -restore [-remote]]
```

When used with `-restore, the -immediate` option deactivates the SRDF/A session without waiting for the two cycle switches to complete before starting the `failover -restore` operation.

To perform a swap resulting in a cascaded R21 device, your Symmetrix arrays must be running Enginuity 5773 and higher. To perform a swap resulting in a concurrent R2 device, your Symmetrix arrays must be running Enginuity 5773.150 and higher.

The `failover -restore` operation has the following restrictions:

- If an SRDF group being failed over is operating in asynchronous mode, then all devices in the group must be failed over in the same operation.
- The R1 and the R2 devices in the failover must be dynamic SRDF devices.
- The R2 device cannot be larger than its R1 device.
- The SRDF swap cannot result in a cascaded R21<-->R21 device pair.
- Not supported by any device group operations with more than one SRDF group.
- Cannot execute this command on both mirrors of a concurrent R2 device (composite group operation). This swap would convert the concurrent R2 into a concurrent R1, with a restore on both mirrors of that concurrent R1.

CHAPTER 4
SRDF/Asynchronous Mode

This chapter explains how to create and manage SRDF/A configurations using the SYMCLI. Topics include:

- Overview 114
- SRDF/Asynchronous operations 116

Overview

Symmetrix arrays support SRDF/Asynchronous (SRDF/A) mode for SRDF devices. Asynchronous mode provides a dependent write consistent copy on the target (R2) device, which is only slightly behind the source (R1) device. SRDF/A session data is transferred to the remote Symmetrix array in predefined timed cycles or *delta sets*, which minimizes the redundancy of same track changes being transferred over the link.

SRDF/A provides a long-distance replication solution with minimal impact on performance that preserves data consistency with the database. This level of protection is intended for backup environments that always need a restartable copy of data at the R2 site. In the event of a disaster at the R1 site or if SRDF links are lost during data transfer, a partial delta set of data can be discarded, preserving consistency on the R2 with a maximum data loss of two SRDF/A cycles or less.

For a description of each of the various SRDF modes, refer to "Understanding and setting SRDF modes of operation" on page 32.

SRDF/A benefits and features

SRDF/A mode provides the following benefits and features:

- Lower operational cost for long-distance data replication with database consistency.
- More efficient link utilization resulting in lower link bandwidth.
- Maintains a dependent write consistent copy on the R2 devices at all times.
- Supports all current SRDF topologies, including point-to-point and switched fabric.
- Requires no additional hardware, such as switches or routers.
- Operates at any given distance without adding response time to the R1 host.
- Supports all hosts and data emulation types supported by the Symmetrix array (such as FBA, CKD, AS 400).
- Minimizes the impact imposed on the back-end DA directors.
- Provides a performance response time equivalent to writing to local non-SRDF devices.
- Allows restore, failover, and failback capability between the R1 and the R2 sites.
- Multiple SRDF/A sessions are allowed per Symmetrix array, with all 250 SRDF groups being SRDF/A-capable.
- SRDF/A-capable devices can be added to composite groups and enabled for database consistency protection using Multi-session Consistency (MSC). Refer to "SRDF consistency group operations" on page 202 for additional information.
- Mode transition from asynchronous to synchronous ensuring database consistency for data on the R2 side is available for devices managed by a device group since. Refer to "Transitioning to synchronous mode" on page 127.
- SRDF/A DSE pools to extend the available cache space for SRDF/A sessions. Refer to "Managing SRDF/A Delta Set Extension pools" on page 128.
- SRDF/A write pacing dynamically prevents conditions that cause cache overflow on both R1 and R2. Refer to "Using SRDF/A write pacing" on page 133.

- You can dynamically change some SRDF/A parameter settings using the `symconfigure` command. Configurable parameters include: session priority, minimum cycle time, enabling SRDF/A transmit idle and SRDF/A DSE pools. Refer to the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide* for details on setting these parameters.
- SRDF/A Reserve Capacity enhances SRDF/A's ability to maintain an operational state when encountering network resource shortfalls that would have previously suspended SRDF/A operations. With SRDF/A Reserve Capacity functions enabled, additional resource allocation can be applied to address temporary workload peaks, periods of network congestion, or even temporary network outages. The two functions that implement SRDF/A Reserve Capacity are Delta Set Extension (DSE) pools, and the Transmit Idle support. They work together to maximize availability of continuous remote replication operations while minimizing operational overhead. Refer to "Managing SRDF/A Delta Set Extension pools" on page 128 and "Using transmit idle" on page 133.
- SRDF/A device pairs can be dynamically added and removed from an active SRDF/A session using the consistency exempt option (`-cons_exempt`). Refer to "Using the consistency exempt option" on page 120.
- Concurrent SRDF/A can asynchronously mirror to recovery sites located at extended distances from the workload site. Refer to "Concurrent SRDF Operations" on page 141.
- SRDF software compression minimizes the amount of data transmitted over an SRDF link using the `symconfigure` command (starting with Enginuity 5875).

SRDF/A restrictions

The restrictions and limitations for running SRDF/A capable devices in asynchronous mode are as follows:

General

- All SRDF/A-capable devices running in asynchronous mode must be managed together by device group, composite group, or device list in an SRDF/A session.
- If there are tracks owed from the R2 to the R1, it is not recommended to switch to asynchronous mode. The force option is required if there are tracks owed to the R1 device when attempting to make SRDF/A-capable devices in asynchronous mode Ready on the link.
- SRDF/A-capable devices that are enabled for consistency group protection must be disabled before attempting to change the mode from asynchronous.
- Symmetrix SRDF Automated Replication (SRDF/AR) control operations are currently **not** supported for SRDF/A-capable devices running in asynchronous mode.

Device groups

The following restrictions apply to device groups for SRDF/A-capable devices:

- Existing RDF1, RDF2, and RDF21 device groups can be used to control SRDF/A-capable devices, but all devices of a certain type (standard, BCV, RBCV, or BRBCV) must be either SRDF/A-capable or non-SRDF/A-capable.
- All SRDF/A-capable devices in the group must be members in the same SRDF/A session within any device type.

Composite group

The following restrictions apply to composite groups for SRDF/A-capable devices:

- All devices within an SRDF group must be in asynchronous mode (or non-asynchronous mode).
- Only standard devices can be managed in asynchronous mode.

TimeFinder snaps and clones

Symmetrix arrays employing either TF/Snap or TF/Clone operations affect whether SRDF devices are allowed to be set in asynchronous mode. Refer to "TimeFinder Snap and Clone Reference" on page 459 for a description of the TF/Snap and TF/Clone pair states and setting SRDF devices to asynchronous mode. Also, for SRDF/A-capable devices operating in asynchronous mode, certain Snap and Clone operations will not be allowed.

For a list of TF/Snap or TF/Clone operations not supported with asynchronous mode, see the *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*.

Move operations

After a `movepair` or `half_movepair` action, the resulting SRDF mode for the moved device will be as follows:

- When moving a device *to* an SRDF group that is currently in asynchronous mode, the resulting SRDF mode for the device being moved will be asynchronous.
- When moving a device *from* an SRDF group in asynchronous mode, the resulting SRDF mode for the device being moved is synchronous.
- When asynchronous mode is not involved, the mode of the device does not change.

SRDF/Asynchronous operations

This section explains the operations you can perform on a configuration operating in SRDF/A mode.

SRDF/A session monitoring

An SRDF/A session consists of a group of devices that were set to asynchronous mode. The SRDF/A session can be monitored using the `symstat` command. Refer to the *EMC Solutions Enabler Array Management CLI Product Guide* for instructions on using the `symstat` command to obtain various Symmetrix performance statistics for monitoring an SRDF/A session.

SRDF/A ordered-write processing

Symmetrix arrays implement asynchronous mode host writes from the source to the target using predetermined timed cycles (called delta sets).

This is different from traditional ordered-write processing.

In asynchronous mode, each delta set contains groups of I/Os for processing. Using cycles of operation, SRDF/A transfers sets of data, one cycle at a time between the R1 and the R2. If the same track is written to more than one time within an active set, SRDF/A will send the update over the link only once. This approach lowers the link bandwidth compared with ordered write-processing, which transfers each write separately. Figure 14 provides a depiction of how SRDF/A delta sets are captured and transferred between sites.

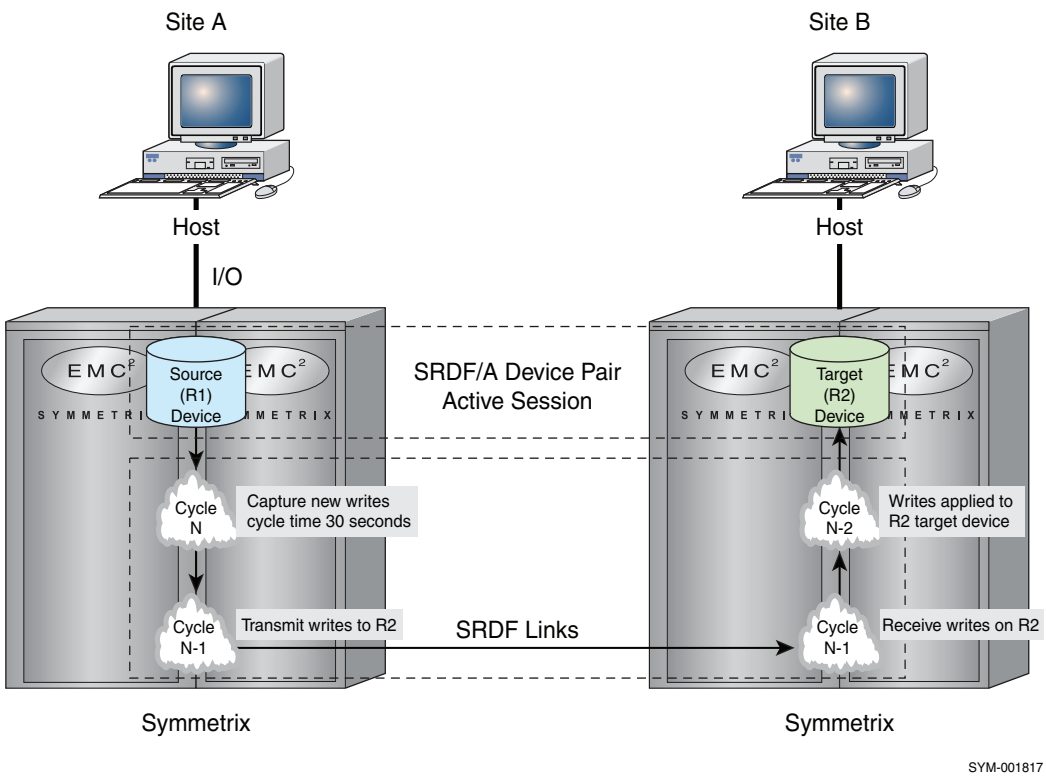


Figure 14 SRDF/Asynchronous mode

Dependent write consistency

A dependent write is a write that is not issued by the application unless some prior I/O has completed. For example, when a DBMS updates a database:

1. It first writes to the disk containing the log to indicate an update is coming, then
2. It writes the data to the database dataset*, and finally
3. It writes again to the log volume to indicate that the database update is complete.

*The write to update the database could be composed of multiple write I/Os. Individual writes can span meta members or multiple tracks.

These three writes (log, database, and log again) are related and each successive I/O is not issued until the prior I/O has successfully completed.

Dependent write consistency uses ordered SRDF/A delta sets between the source (R1) and the target (R2) in cycles, as shown in Figure 14. Dependent write consistency ensures that all writes to the R2 are processed in sequential numbered sets to maintain a consistent copy of data between the R1 and the R2.

When the first SRDF/A cycle (*N*) is active, it collects any new writes on the R1, overwriting any duplicate tracks intended for data transfer over the link. The cycle is active for a predetermined amount of time, which can be configured on the Symmetrix array (default is 30 seconds). After the set time has been reached, the delta set data moves into the next cycle position (*N-1*) and begins transferring the delta set over the link to the R2. A new cycle *N* then begins collecting new writes again for the next delta set transfer.

In cycle *N-1*, the delta set is temporarily collected on the R2 side for destaging. When the *N-1* cycle has finished transferring data into the R2 and the minimum cycle time has elapsed, the delta set data moves into the next cycle position (*N-2*) and begins destaging the data to the R2 storage devices. The delta set data is considered committed to the R2 in cycle *N-2* as it is applied to disk.

One delta set is dependent upon the other for achieving write consistency. No cycle can begin until the prior one has completed. All data is transferred at the block level.

Note: The cycle is elongated if the write transfer or destaging exceeds the set cycle time.

When all delta set *N-2* data is applied to the R2 target device, the R1 and R2 are considered to be in the *consistent pair state*, both containing a consistent image of data. The user can verify if the data in a current SRDF/A session has been applied to the R2 by using the `symrdf checkpoint` command. Refer to "Confirming the R2 data copy" on page 126.

Consistency can be enabled and disabled for SRDF/A-capable devices. For more information, refer to "Enable and disable consistency protection for SRDF/A devices" on page 59.

For instructions on how to enable consistency for composite group operations, refer to "SRDF consistency group operations" on page 202.

Setting SRDF/Asynchronous mode

To remotely mirror pairs in the `prod` device group in asynchronous mode, type:

```
symrdf -g prod set mode async
```

Optionally, you can set asynchronous mode for devices in a composite group or device file, such as:

```
symrdf -cg CgName set mode async
symrdf -f[ile] FileName set mode async
```

A device status check is performed on all TimeFinder/Snap and TimeFinder/Clone device pairs in the group before the operation is allowed, as described in "TimeFinder Snap and Clone Reference" on page 459.

Setting SRDF/A attributes

Use the following syntax to set the SRDF/A group attributes on an SRDF group:

```
symrdf -sid <SymmID> -rdfg <GrpNum> [-v] [-symforce]
     [-noprompt] [-i <Interval>] [-c <Count>]

.............

set rdfa
   <[-cycle_time <CycleTime>]
   [-priority <SessPriority>]
   [-transmit_idle <State>]>
   [-both_sides]
```

To set these group attributes, the remote side must be reachable.

Where:

- `-cycle_time` (`-cyc`) is the minimum time to wait before attempting an SRDF/A cycle switch. Valid values are 1 through 60 seconds.

  For Symmetrix arrays running Enginuity 5874 and higher, the default is 15.

  For Symmetrix arrays running a version lower than 5874, the default value is 30.

  See "Enabling and disabling SRDF consistency protection" on page 205 for details about how minimum cycle times are handled when you enable consistency protection.

- `-priority` (`-pri`) determines which SRDF/A sessions are dropped if the cache becomes full. Valid values are 1 through 64. The default value is 33.
- `-transmit_idle` (`-tra`) provides an extra layer of protection to ensure an SRDF/A session does not drop when the link cannot transmit data. Valid state values are on (default) and off.
- `-both_sides` applies the SRDF/A attributes to both the source and target sides of an SRDF/A session. Otherwise, these attributes are only applied to the source side.

To locally set the cycle time to 32 and the priority to 20 for SRDF/A group 12, type:

```
symrdf -sid 134 -rdfg 12 set rdfa -cycle_time 32 -priority 20

An RDF Set 'Attributes' operation execution is in progress for RDF group 12.
Please wait...

    SRDF/A Set Min Cycle Time(1134,012)........................Started.
    SRDF/A Set Min Cycle Time (1134,012)........................Done.
    SRDF/A Set Priority (1134,012)..............................Started.
    SRDF/A Set Priority (1134,012)............................,,,,Done.
The RDF Set 'Attributes' operation successfully executed for RDF group 12.
```

You can also use the `symconfigure` command to set these SRDF/A attributes. For more information, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Checking for R1 invalid tracks

The `symrdf verify -consistent` command verifies that SRDF device pairs are in the R2 Consistent pair state, indicating no invalid tracks are owed to the R2 side from its R1 side.

If an SRDF pair is in the Split state and the host writes to its R2 device, invalid tracks are then owed to its R1 device. When this pair is restored, the pair is still considered in the Consistent state because no invalid tracks are owed to the R2 device. It does not recognize invalid tracks owed from R2 to R1. By using the `-noinvalids` option with the `-consistent` option, an additional check is performed to verify whether invalid tracks exist on both the R1 and R2 devices.

The `-noinvalids` (`-noinv`) option must be issued with the `-consistent` option, and applies to device groups, composite groups, and device files.

Use the following example to monitor the clearing of invalid tracks for the `dg1` device group:

```
symrdf verify -g dg1 -consistent -noinv -i 60

None of the devices in the group 'dg1' are in 'Consistent with no
invalid tracks' state.

Not all devices in the group 'dg1' are in 'Consistent with no invalid
tracks' state.

All devices in the group 'dg1' are in 'Consistent with no invalid
tracks' state.
```

For example outputs of this command, see "Verifying if invalid tracks are owed to R1 from R2" on page 373.

Using the consistency exempt option

The consistency exempt option (`-cons_exempt`) provides the ability to dynamically add and remove device pairs from an active SRDF/A session without affecting the state of the session, or the reporting of SRDF pair states for devices that are not the target of the operation.

The consistency exempt option is provided for an SRDF/A environment to indicate that the devices should be temporarily considered exempt from the consistency requirements of the group. This allows the group to be dynamically expanded or decreased without deactivating the SRDF/A session. When adding or removing device pairs in an active session, devices can be placed in a consistency exempt state and are excluded from the group's consistency check. The consistency exempt state is automatically removed after the devices are resumed and fully synchronized and two full cycle switches have occurred, or after the devices are removed from the group.

The `-cons_exempt` option can be used with the `createpair`, `movepair`, `half_movepair`, and `suspend` commands. When `-cons_exempt` is used with `movepair` and `half_movepair`, the controlled devices are set consistency exempt within the new SRDF/A group. Additionally, devices that are currently suspended and consistency exempt within an active SRDF/A session can be controlled apart from other devices in the session for `resume`, `establish`, `deletepair`, `half_deletepair`, `movepair`, and `half_movepair` operations.

Note: The consistency exempt option (-cons_exempt) is available with Symmetrix arrays running Enginuity 5773.150 and higher.

The following restrictions apply for using the consistency exempt option:

- The consistency exempt option cannot be used on devices that are part of an SRDF/Star configuration.
- The consistency exempt option cannot be used on an SRDF/A session that is in the Transmit Idle state.
- If the device is an R2 device and the SRDF/A session is active, the `half_movepair` and `half_deletepair` commands are blocked.
- If the session is deactivated before the consistency exempt state is cleared, when re-activated, the device remains in the consistency exempt state until the device has no invalid tracks that need to be synchronized.
- A movepair operation of an SRDF pair to another SRDF group with an active SRDF/A session is only allowed when the SRDF pair state is suspended and can be blocked if in the failed over or split pair state.
- The `createpair` and `movepair` operations are allowed without the `-con_exempt` option if the new SRDF group is operating in the asynchronous mode but the SRDF/A session is not active.

Adding device pairs to an active SRDF/A session

In the following example, device pairs are added to an active SRDF/A session using a device file called `MyFile`:

1. Use the `createpair` command to add a new device pair to a temporary SRDF group and synchronize it. In this example, the devices synchronize faster due to a a higher throughput of a synchronous link:

   ```
   symrdf createpair -file Myfile -sid 1234 -rdfg 10 -type RDF1
   -establish
   ```

   ```
   symrdf verify -file MyFile -sid 1234 -rdfg 10 -synchronized
   ```

2. Suspend the device pair so that it can be moved into the SRDF/A group:

   ```
   symrdf suspend -file MyFile -sid 1234 -rdfg 10
   ```

   Note: Solutions Enabler does not accept the consistency exempt (`-cons_exempt`) flag with any group operating in synchronous mode. For example, if you issued this flag with the above command, it would fail because `-rdfg 10` is synchronous.

3. Move the device pair from the current SRDF group into the active SRDF/A group:

   ```
   symrdf movepair -file MyFile -sid 1234 -rdfg 10 -new_rdfg 20
   -cons_exempt
   ```

4. Resume the device pair:

   ```
   symrdf resume -file MyFile -sid 1234 -rdfg 20
   ```

5. Verify when the device pair becomes consistent and there are no invalid tracks on the R1 and R2 sides:

   ```
   symrdf verify -file MyFile -sid 1234 -rdfg 20 -consistent -noinvalids
   ```

   Note: The application must not use the R1 device until the SRDF pair state is Consistent.

Removing device pairs from an active SRDF/A session

In the following example, device pairs are removed from an active SRDF/A session using a device file called `MyFile`:

1. Suspend the device pair so that it can be moved into another SRDF group:

   `symrdf suspend -file MyFile -sid 1234 -rdfg 20 -cons_exempt`

2. Move the device pair from the current SRDF group into the another SRDF group:

   `symrdf movepair -file MyFile -sid 1234 -rdfg 20 -new_rdfg 30`

3. Resume the devices:

   `symrdf resume -file MyFile -sid 1234 -rdfg 30`

4. Verify when the device pair becomes consistent:

   `symrdf verify -file MyFile -sid 1234 -rdfg 30 -synchronized`

   Note: The application must not use the R1 device until the SRDF pair state is Consistent.

Displaying SRDF/A session status

When asynchronous mode is set for a group of devices, the SRDF/A-capable devices in the group are considered part of the SRDF/A session. The session status is displayed as active or inactive, as follows:

- Active indicates the SRDF/A mode is activated and that SRDF/A session data is currently being transmitted in operational cycles to the R2.
- Inactive indicates the SRDF/A devices are either Ready or Not Ready on the link and working in their basic mode (synchronous, semi-synchronous, or adaptive copy).

Note: If the links are suspended or a split operation is in process, SRDF/A is disabled and will show a session status of Inactive.

Use the `symdg show` command to display SRDF/A session status information:

```
symdg show dgr21

Group Name:  dgr21

    Group Type                                    : RDF21
    Device Group in GNS                           : No
    Valid                                         : Yes
    Symmetrix ID                                  : 000187940256
    Group Creation Time                           : Wed Feb 4 9:5:1 2010
    Vendor ID                                     : EMC Corp
    Application ID                                : SYMCLI

    Number of STD Devices in Group                :    0
    Number of Associated GK's                     :    0
    Number of Locally-associated BCV's            :    0
    Number of Locally-associated VDEV's           :    0
    Number of Locally-associated TGT's            :    0
    Number of Remotely-associated VDEV's(STD RDF):    1
    Number of Remotely-associated BCV's (STD RDF):    0
    Number of Remotely-associated TGT's(TGT RDF) :    0
    Number of Remotely-associated BCV's (BCV RDF):    0
```

```
    Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
    Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
    Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
    Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0

    VDEV Devices Remotely-associated (STD RDF) (1):
        {
        ----------------------------------------------------------------
                                                Sym                Cap
        LdevName             PdevName           Dev  Att. Sts      (MB)
        ----------------------------------------------------------------
        RVDEV001             N/A            0171          NR      2063
        }

    Device Group RDF Information
        {
Device Group RDF Information
        {
        RDF Type                                : R1
        RDF (RA) Group Number                   : 153                (98)

        Remote Symmetrix ID                     : 000194900311

        R2 Device Is Larger Than The R1 Device : False

        Paired with a Diskless Device           : False
        Paired with a Concurrent Device         : False
        Paired with a Cascaded Device           : False
        Thick Thin Relationship                 : False

        RDF Pair Configuration                  : Normal
        RDF STAR Mode                           : False

        RDF Mode                                : Asynchronous
        RDF Adaptive Copy                       : Disabled
        RDF Adaptive Copy Write Pending State   : N/A
        RDF Adaptive Copy Skew (Tracks)         : 32767

        RDF Device Domino                       : Disabled

        RDF Link Configuration                  : Fibre
        RDF Link Domino                         : Disabled
        Prevent Automatic RDF Link Recovery     : Enabled
        Prevent RAs Online Upon Power ON        : Enabled

        Device RDF Status                       : Ready           (RW)

        Device RA Status                        : Ready           (RW)
        Device Link Status                      : Not Ready       (NR)
        Time of Last Device Link Status Change : N/A

        Device Suspend State                    : Offline
        Device Consistency State                : Disabled
        Device Consistency Exempt State         : Disabled
        RDF R2 Not Ready If Invalid             : Disabled
        Device Write Pacing Exempt State        : Disabled
        Effective Write Pacing Exempt State     : Disabled

        Device RDF State                        : Ready           (RW)
        Remote Device RDF State                 : Write Disabled  (WD)

        RDF Pair State ( R1 <-  -> R2 )         : Suspended

        Number of R1 Invalid Tracks             : 0
        Number of R2 Invalid Tracks             : 0
```

Listing SRDF/A device information

Use the `symrdf list -rdfa` command to list SRDF/A-capable devices (R1, R2 and R21 devices) that are currently configured in SRDF groups.

Note: SRDF/A-capable does not mean the device is actually operating in asynchronous mode, only that it is capable of doing so. There is no command that lists devices that are actually operating in asynchronous mode.

The device type is shown as *R1* for SRDF/A-capable devices on the R1 and type *R2* for SRDF/A-capable devices on the R2. The *R21* device type represents a cascaded SRDF device configuration. See Chapter 6, "Cascaded SRDF Operations," for more information about R21 devices.

The following is example output of the `symrdf list -rdfa` command:

```
symrdf -sid 321 list -rdfa

Symmetrix ID: 000192600321

                              Local Device View
---------------------------------------------------------------------------
                     STATUS     MODES                      RDF  S T A T E S
Sym        RDF       ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G     SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------   ---------  ----- -------  ------- --- ---- -------------

0060 0060   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0061 0061   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0062 0062   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0063 0063   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0064 0064   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0065 0065   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0066 0066   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0067 0067   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E0 07E0   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E1 07E1   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E2 07E2   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E3 07E3   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E4 07E4   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E5 07E5   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E6 07E6   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E7 07E7   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E8 07E8   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E9 07E9   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EA 07EA   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EB 07EB   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EC 07EC   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07ED 07ED   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EE 07EE   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EF 07EF   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0800 0800   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0801 0801   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0802 0802   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0803 0803   R1:128   RW RW RW   C.D1.       0        0 RW  WD   Synchronized

Total                                 -------- --------
  Track(s)                                   0        0
  MB(s)                                    0.0      0.0

Legend for MODES:

 M(ode of Operation)    : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
```

```
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Mirror) T(ype)       : 1 = R1, 2 = R2
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

The following output of the `symrdf query` command displays SRDF/A group information. In this display, the RDFA session is inactive and the SRDF pairs are in a Suspended state.

symrdf -g AsyncGrp1 query -rdfa

```
Device Group (DG) Name             : AsyncGrp1
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)

RDFA Session Number                : 127
RDFA Cycle Number                  : 0
RDFA Session Status                : Inactive
RDFA Consistency Exempt Devices    : No
RDFA Minimum Cycle Time            : 00:00:30
RDFA Avg Cycle Time                : 00:00:00
Duration of Last cycle             : 00:00:00
RDFA Session Priority              : 33
Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1          : 00:00:54
R2 Image Capture Time              : Fri Jul 23 15:05:53 2010
R2 Data is Consistent              : True
RDFA R1 Side Percent Cache In Use  :  0
RDFA R2 Side Percent Cache In Use  :  0
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
Transmit Idle Time                 : 00:00:00
R1 Side Shared Tracks              : 0

Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDACE STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 WD       0        0 A..X. Suspended
DEV002  0061 RW       0        0 NR 0061 WD       0        0 A..X. Suspended
DEV003  0062 RW       0        0 NR 0062 WD       0        0 A..X. Suspended
DEV004  0063 RW       0        0 NR 0063 WD       0        0 A..X. Suspended
DEV005  0064 RW       0        0 NR 0064 WD       0        0 A..X. Suspended
DEV006  0065 RW       0        0 NR 0065 WD       0        0 A..X. Suspended
DEV007  0066 RW       0        0 NR 0066 WD       0        0 A..X. Suspended
DEV008  0067 RW       0        0 NR 0067 WD       0        0 A..X. Suspended
DEV009  07E0 RW       0        0 NR 07E0 WD       0        0 A..X. Suspended
DEV010  07E1 RW       0        0 NR 07E1 WD       0        0 A..X. Suspended
DEV011  07E2 RW       0        0 NR 07E2 WD       0        0 A..X. Suspended
DEV012  07E3 RW       0        0 NR 07E3 WD       0        0 A..X. Suspended
DEV013  07E4 RW       0        0 NR 07E4 WD       0        0 A..X. Suspended
DEV014  07E5 RW       0        0 NR 07E5 WD       0        0 A..X. Suspended
DEV015  07E6 RW       0        0 NR 07E6 WD       0        0 A..X. Suspended
DEV016  07E7 RW       0        0 NR 07E7 WD       0        0 A..X. Suspended
DEV017  07E8 RW       0        0 NR 07E8 WD       0        0 A..X. Suspended
DEV018  07E9 RW       0        0 NR 07E9 WD       0        0 A..X. Suspended
DEV019  07EA RW       0        0 NR 07EA WD       0        0 A..X. Suspended
DEV020  07EB RW       0        0 NR 07EB WD       0        0 A..X. Suspended
DEV021  07EC RW       0        0 NR 07EC WD       0        0 A..X. Suspended
```

```
Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

Using the immediate option

The `-immediate` option applies to devices participating in an active SRDF/A session. When used with the `failover, split,` or `suspend` commands, the SRDF/A session will be immediately dropped and the devices will be made Not Ready on the link. The devices will remain in asynchronous mode and any pending tracks will be converted to invalid tracks.

Using the `-immediate` option may result in remote invalid tracks on both the R1 and the R2 sides. The `-immediate` option does not compromise the consistency of data on the R2 side, but requires operator intervention to resolve any invalid tracks by using the correct `symrdf` command and pair state, as described in Table 55 on page 498.

By default, issuing a `failover`, `split`, or `suspend` command without the `-immediate` option will cause the SRDF/A session to be dropped and the devices to be made Not Ready on the link at the end of the current cycle. Execution time of the command may be elongated, but yields no remote invalid tracks on the R2 side.

The `symrdf query -rdfa` command displays the number of tracks not committed to the R2 side as well as any invalid tracks.

Note: If consistency is enabled on SRDF/A-capable devices within the group, then the `-force` option must be applied for the `failover`, `split`, and `suspend` commands.

Using BCVs to preserve the R2 SRDF/A data copy

Although not required for asynchronous mode, it is recommended that you use TimeFinder BCVs at the remote site to mirror R2 devices and preserve a consistent image of data before resynchronization operations. R2 device BCVs can be consistently split off of the R2 without having to drop the SRDF links and without disruption to the SRDF/A operational cycles. R2 BCVs can be controlled from the R1-side or the R2-side host as long as the device groups have been defined on that host. Controlling the R2 BCVs from the R1-side host requires using the `symmir` command with the `-rdf` option.

For example, to consistently split off the R2 BCVs in group `prod` from the R1 host, type:

```
symmir -g prod split -rdf -consistent
```

Note: For more information on the `symmir -consistent split` command, refer to the *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*.

Confirming the R2 data copy

The `checkpoint` action confirms to the caller that data written in the current SRDF/A cycle has been successfully committed to the R2. The option is only valid for SRDF/A capable devices participating in an active SRDF/A session. All supplied devices must be in the same SRDF/A session.

Optionally, you can target devices in a device group, composite group, or device list. The syntax for a checkpoint operation using a file is:

```
symrdf -file Filename -sid SymmID -rdfg GrpNum
          [-offline] [-i Interval] [-c Count] checkpoint
```

The syntax for a `checkpoint` operation using a CG is:

```
symrdf -cg CgName [-i Interval] [-c Count]
     [ -hop2 ]
   [-rdfg <SymmID:GrpNum | name:RdfGroupName>]

checkpoint
```

The syntax for a `checkpoint` operation using a DG is:

```
symrdf -g DgName [-i Interval] [-c Count] [-rdfg GrpNum]
[ -hop2 | -bcv [-hop2] | -all | -rbcv | -brbcv]

  checkpoint
```

When using an RDF21 CG or DG, the `checkpoint` command can only be issued against one SRDF group at a time. For a checkpoint using a device group, the default is to perform the checkpoint action against the first link (R1 -> R21 pair). To perform a checkpoint operation for the other link (R21 -> R2 pair), the `-rdfg` parameter must be specified. When using an RDF21 CG, specify `-rdfg` or `-rdfg_name`.

When using an RDF1 or RDF2 CG or DG the user must specify the `-hop2` flag to process the `checkpoint` command on the second hop SRDF pair for a cascaded SRDF relationship. The `-hop2` parameter can also be used in conjunction with the `-rdfg` parameter to checkpoint the second hop of the cascaded SRDF devices on the requested SRDF groups or SRDF group name.

Examples

For example, to confirm R2 data copy for asynchronous-mode devices in device group `prod`, type:

```
symrdf -g prod checkpoint
```

To confirm the R2 data copy for SRDF/A-capable devices in device group `Test` in RA group 7 on the second hop of a cascaded SRDF configuration, type:

```
symrdf -g Test -rdfg 7 -hop2 checkpoint
```

Transitioning to synchronous mode

Solutions Enabler supports a consistent mode transition from asynchronous to synchronous for devices managed by a device group or device file. A consistent mode transition preserves database consistency for data on the R2 side.

This functionality is supported on Symmetrix arrays running Enginuity 5671 and later.

For example, to switch modes from asynchronous to synchronous and maintain R2 data consistency in group `prod`, type:

```
symrdf -g prod -consistent set mode sync
```

To switch modes from asynchronous to synchronous and maintain R2 data consistency for devices listed in device file `devfile1`, type:

```
symrdf -f devfile1 -consistent set mode sync
```

Note: Completion of a consistent mode transition requires two SRDF/A cycle switches.

Enabling consistency for concurrent SRDF/A devices

Consistency can be enabled for concurrent SRDF/A device pairs in a file or a device group. In a concurrent relationship, each R2 mirror must be enabled separately.

Device file

The following command enables the device pairs listed in `file1` with SRDF mirrors in SRDF group 55 for consistency. All device pairs in that SRDF group must be in the device file.

If `file1` includes concurrent devices, only the target mirror in SRDF group 55 is enabled. This command does not enable the other target mirror in the concurrent relationship, which is in the SRDF group 56.

```
symrdf -f file1 -sid 123 -rdfg 55 enable
```

Device group

The following command enables the target mirror in SRDF group 56 of `devgroup2` for consistency. All device pairs in that SRDF group must be in the device group.

If `devgroup2` includes concurrent devices, only the target mirror in SRDF group 56 is enabled. This command does not enable the other target mirror in the concurrent relationship, which is in the SRDF group 55.

```
symrdf -g devgroup2 -rdfg 56 enable
```

Because you must enable each R2 mirror separately in a concurrent relationship for consistency, you cannot use the `-rdfg all` option. For example, to enable consistency for target mirrors in SRDF group 56 and SRDF group 57 of `devgroup2`, specify these individual commands:

```
symrdf -g devgroup2 -rdfg 56 enable
symrdf -g devgroup2 -rdfg 57 enable
```

Managing SRDF/A Delta Set Extension pools

When many SRDF/A groups run in parallel in the same Symmetrix array, complex I/O profiles and potential interruptions in link availability and bandwidth make the task of calculating reasonable cache requirements difficult. The SRDF/A Reserve Capacity feature, Delta Set Extension (SRDF/A DSE), extends the cache space available for SRDF/A session cycles by off loading some or all of its cycle data from cache to preconfigured disk storage, or pools, which are similar to SAVE device pools.

SRDF/A DSE pools provide a safety net for running SRDF/A sessions; however, SRDF/A continues to operate normally with this functionality disabled.

Note: It is still possible for an SRDF/A session to stop if all of the SRDF/A DSE pool cache is used.

Follow these guidelines when using SRDF/A DSE:

- Pools are made up of SAVE devices that are specifically configured to be used for pools.
- A special pool DEFAULT_POOL exists by default and contains SAVE devices that have not been assigned to any named pools, but are available for operations. The devices in the DEFAULT_POOL are either enabled or disabled, but have used tracks.
- An SRDF/A DSE pool cannot have the same name as a Snap pool for the same Symmetrix array.
- An SRDF/A DSE pool can only contain one type of device emulation, either FBA, CKD3390, CKD3380, or AS400.
- SRDF/A DSE pools can be shared among multiple SRDF/A groups.
- A single SRDF/A group can have up to 4 SRDF/A DSE pools associated with it (one for each device emulation type).

Setting SRDF/A DSE attributes

Use the following syntax to set the SRDF/A DSE attributes for an SRDF group:

```
symrdf -sid <SymmID> -rdfg <GrpNum> [-v] [-symforce]
     [-noprompt] [-i <Interval>] [-c <Count>]

.............

set rdfa_dse
   <[-autostart <State>]
   [-threshold <DseThreshold>]
   [-fba_pool <PoolName>]
   [-ckd3390_pool <PoolName>]
   [-ckd3380_pool <PoolName>]
   [-as400_pool <PoolName>]>
   [-both_sides]>
```

To set these group attributes, the remote side must be reachable.

Where:

- `-autostart` (`-aut`) specifies whether SRDF/A DSE is automatically enabled or disabled when an SRDF/A session is activated for an SRDF group. Valid state values are on or off (default).
- `-threshold` (`-thr`) specifies the percentage of the Symmetrix array's write pending limit. Once the cache usage of all active SRDF/A groups in the array exceeds this limit, data tracks for this SRDF group start to spill over to disks. Valid values are between 20 and 100. The default is 50.
- `-fba_pool` (`-fba`) indicates the DSE pool name containing SAVE devices with FBA emulation. If no argument is provided, the currently associated FBA pool is removed from the group.

- `-ckd3380_pool` (`-ckd3380`) indicates the DSE pool name containing SAVE devices with CKD 3380 emulation. If no argument is provided, the currently associated CKD 3380 pool is removed from the group.
- `-ckd3390_pool` (`-ckd3380`) indicates the DSE pool name containing SAVE devices with CKD 3390 emulation. If no argument is provided, the currently associated CKD 3390 pool is removed from the group.
- `-as400_pool` (`-as400`) indicates the DSE pool name containing SAVE devices with an AS400 emulation. If no argument is provided, the currently associated AS400 pool is removed from the SRDF group.
- `-both_sides` sets the SRDF/A DSE attributes on both the source and target sides of an SRDF/A session. Otherwise, these attributes are only set on the source side.

You can also use the `symconfigure` command to set these SRDF/A DSE attributes. For more information, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Example

To clear the DSE pool names, type:

```
symrdf -sid 432 -rdfg 75 set rdfa_dse -fba_pool -ckd3390_pool -ckd3380_pool
   -as400_pool

An RDF Set 'Attributes' operation execution is in progress for
RDF group 75. Please wait...

    SRDF/A Set FBA Pool (0432,075)....................................Started.
    SRDF/A Set FBA Pool (0432,075)....................................Done.
    SRDF/A Set CKD3380 Pool (0432,075)................................Started.
    SRDF/A Set CKD3380 Pool (0432,075)................................Done.
    SRDF/A Set CKD3390 Pool (0432,075)................................Started.
    SRDF/A Set CKD3390 Pool (0432,075)................................Done.
    SRDF/A Set AS400 Pool (0432,075)..................................Started.
    SRDF/A Set AS400 Pool (0432,075)..................................Done.

The RDF "Attributes'' operation successfully executed for RDF group 75.
```

Adding devices to an SRDF/A DSE pool

Devices can be added to a pool if they are disabled, inactive, and do not belong to another pool. When adding devices to the pool, they can be enabled using the following form:

```
add dev SymDevName[:SymDevName] to pool PoolName
type = <snap | rdfa_dse>
[, member_state = <ENABLE | DISABLE> ];
```

For example:

```
add dev 018B:018C to pool finance,
type = rdfa_dse,
member_state=ENABLE;
```

SRDF/A DSE with thin devices is supported on:

- Symmetrix arrays running Enginuity 5773.150 and higher, and
- Symmetrix arrays running Enginuity 5874.228.182 and higher.

Removing devices from an SRDF/A DSE pool

SAVE devices can be removed from an SRDF/A DSE pool, but they must be disabled and drained. Drained devices become inactive within their pool and can then be moved out of the pool. When you remove a device from a pool, it becomes available for use by other SAVE device pools.

To remove a SAVE device from an SRDF/A DSE pool, use the following form:

```
remove dev SymDevName[:SymDevName]
from pool PoolName,
type = <snap | rdfa_dse>;
```

For example:

```
remove dev 018B from pool finance, type = rdfa_dse;
```

The last device cannot be removed from an SRDF/A DSE pool if the pool is currently associated with an SRDF group.

Enabling and disabling devices in an SRDF/A DSE pool

Devices in an SRDF/A DSE pool can be enabled or disabled. The devices in the pool do not all have to be in the same state (enabled or disabled). If all the devices in a pool are disabled, the pool is disabled. If at least one device in a pool is enabled, then the pool is enabled. To enable or disable a range of devices, all the devices must be in the same pool.

To enable or disable a device an SRDF/A DSE pool, use the following form:

```
enable dev SymDevName[:SymDevName] in pool PoolName,
type = <snap | rdfa_dse>;
```

For example:

```
enable dev 018C in pool finance,
type = rdfa_dse
```

You cannot disable all the devices in an SRDF/A DSE pool if the pool is currently associated with an SRDF group and SRDF/A DSE is active for the group.

Associating an SRDF group with a DSE pool

To set the SRDF/A DSE threshold, associate an SRDF group with a pool, and activate DSE automatically, type:

```
symconfigure commit -sid 12 -file setup_dse.cmd
```

where `setup_dse.cmd` contains:

```
set rdf group 7 rdfa_dse_threshold=20;
set rdf group 7 rdfa_dse_pool=r1pool, emulation=fba;
set rdf group 7 rdfa_dse_autostart=enable;
```

Note: Place the command for setting the threshold first in the file.

SRDF/A DSE pools are created and managed in command files that are executed using the `symconfigure` command. For more information about using `symconfigure`, refer to the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Monitoring SRDF/A DSE pool usage

The `symrdf monitor` command monitors SAVE devices' usage in an SRDF/A DSE pool. This command checks the total percent full of the SAVE devices currently configured in an SRDF/A DSE pool and can optionally execute a script file if a specified percentage is encountered. The following is the command syntax:

```
symrdf [-sid SymmID] [-i Interval] [-c Count] [-offline]
       [-percent <1-100> -action ScriptFile [-norepeat]
   monitor -svp PoolName
```

Where:

- `-percent` causes the action script to be executed when the percent-full value is encountered.
- `-action` selects a script that should be run when the specified percent value is encountered. The full path name to the action script needs to be specified. The first argument passed to the script is automatically set to the percent value.
- `-norepeat` specifies that the action script should only be run once if the threshold has been met. Used with the action script option on the monitor command.
- `-svp` specifies the SRDF/A DSE save pool name.

The display for the `symrdf monitor` command lists all the SAVE devices' usage in the specified pool, as shown in the following example command and output:

```
symrdf monitor -percent 40 -action a.ksh -i 120 -c 3 -svp finance

Symmetrix ID: 000287400166

                R D F A   D S E  S A V E   D E V I C E S
-------------------------------------------------------------------
      Device     SaveDevice        Total      Used      Free     Full
Sym   Emulation  Pool Name        Tracks    Tracks    Tracks      (%)
-------------------------------------------------------------------
018B  FBA        finance           66000         0     66000        0
018C  FBA        finance           66000         0     66000        0

Total                           --------- --------- ---------     ----
  Tracks                           132000         0    132000        0
  MB(s)                            4125.0       0.0    4125.0
```

Querying SRDF/A DSE status

Use the `symrdf query` command to display the SRDF/A DSE status information.

To query the SRDF/DSE status for `Rdf1Grp`, type:

```
symrdf query -g Rdf1Grp -rdfa
```

Activating and deactivating SRDF/A DSE

For information on activating and deactivating SRDF/A DSE, see "Command option descriptions" on page 52.

Using transmit idle

The transmit idle feature provides an extra layer of protection to ensure an SRDF/A session does not drop when the link cannot transmit data. If a link fails when transmit idle is enabled, the SRDF pair enters the Transmit Idle state after the link limbo time expires. When the link recovers, the SRDF pair state returns to Consistent or SyncInProg. For information on link limbo, see "Specifying a link limbo for a dynamic group" on page 97.

When the SRDF pair is in the Transmit Idle state, only the following operations are allowed from the R1 side:

- `rw_enable -r1`
- `write_disable -r1`
- `Ready -r1`
- `not_ready -r1`
- `suspend -immediate`

When the SRDF pair is in the Transmit Idle state, only the following operations are allowed from the R2 side:

- `suspend -immediate`
- `failover -immediate`

To set transmit idle on the source and target sides for SRDF/A group 12, type:

```
symrdf -sid 134 -rdfg 12 set rdfa -tra on -both_sides

An RDF Set 'Attributes' operation execution is in progress for RDF group 12.
Please wait...

    SRDF/A Set Transmit Idle (1134,012)..........................Started.
    SRDF/A Set Transmit Idle (1134,012)..........................Done.
    SRDF/A Set Transmit Idle (0601,015)..........................Started.
    SRDF/A Set Transmit Idle (0601,015)..........................Done.
The RDF Set 'Attributes' operation successfully executed for RDF group 12.
```

Note: If at the beginning of a control action, all SRDF/A groups are not in the Transmit Idle state, the action fails if one of the groups enters the Transmit Idle state during processing.

You can also use the `symconfigure` command to set transmit idle. For more information, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Using SRDF/A write pacing

The SRDF/A group-level and device-level write pacing feature helps secure the availability of an SRDF/A session by preventing conditions that cause cache overflow on both the R1 and R2 sides. It can work in conjunction with the SRDF/A DSE and transmit idle functionality.

For information on how to manually activate and deactivate this feature, refer to "Activate and deactivate SRDF/A write pacing" on page 57.

SRDF/A write pacing bases some of its actions on the following:

- R1 side cache usage
- Transfer rate of data from transmit delta set to receive delta set
- Restore rate on the R2 side

Group-level write pacing

SRDF/A group-level write pacing detects when the I/O service rates are lower than the host I/O rates, and then takes corrective actions to slow down the host I/O rates to match the SRDF I/O service rates. When activated, this functionality performs write I/O pacing on all devices in the SRDF group during an SRDF/A session. This operation also provides an exemption capability to prevent group-level write I/O pacing on specified devices within a group.

SRDF/A group-level write pacing also monitors and responds to spikes in the host write I/O rates as well as slowdowns in data transmittal between R1 and R2 and in the R2 restore rates. By monitoring and pacing the host write I/O rates, this feature controls the amount of cache used by SRDF/A. This prevents cache overflow on both the R1 and R2 sides, and helps the SRDF/A session to stay up and running.

Enhanced functionality

Prior to Enginuity 5876, SRDF/A group-level write pacing was concerned solely with managing host I/O rates so they would not overrun the R1->R2 data transmittal rate on the SRDF link.

Enginuity 5876 extends SRDF/A group-level write pacing so that it can respond to the following conditions:

- The write-pending level on an R2 device in an active SRDF/A session reaches the device's write-pending limit.
- The restore (apply) cycle time on the R2 side is longer than the capture cycle time.

As a result, the enhanced group-level write pacing feature can effectively pace host write I/Os in the following operational scenarios:

- Slower restore (apply) cycle times on specific R2 devices that are managed by slower-speed physical drives.
- FAST operations that lead to an imbalance in SRDF/A operations between the R1 and R2 sites.
- Sparing operations that lead to R2-side DAs becoming slower in overall restore operations.
- Production I/Os to the R2 side that lead to DAs and/or RAs becoming slower in restore operations.
- Restore delays during the pre-copy phase of TimeFinder/Clone sessions before activation.

The configuration and management of group-level write pacing are unaffected by this enhancement.

Requirements

- The group-level pacing function is supported on Symmetrix arrays running on Enginuity 5874.207.166 and higher.
- The exemption capability requires that the Symmetrix array on the R1 side be running Enginuity 5875 and higher.
- The enhanced group-level write pacing functionality requires that the R1 and R2 Symmetrix arrays be running Enginuity 5876 and higher. Otherwise, group-level write pacing is available without this enhanced functionality.

Considerations

- Only the group-level pacing values configured for the SRDF group on the R1 side of the SRDF/A session are used.
- In a cascaded SRDF environment:
  - With Enginuity 5876 and lower, group-level write pacing is only supported on the R1->R21 hop of the relationship.
  - With Enginuity 5876 Q4 2012 SR and higher, group-level write pacing is supported on both the R1->R21 and R21->R2 hops of the relationship.
- In a concurrent SRDF/A environment, group-level pacing is supported on both mirrors of the concurrent R1. In this case, write pacing calculations are performed independently for the two SRDF/A sessions, and the host write I/Os sessions are subject to the greater of the two calculated delays.

Device-level write pacing

SRDF/A device-level write pacing addresses conditions that lead to cache overflow specifically due to TimeFinder/Snap and TimeFinder/Clone sessions on an R2 device running in asynchronous mode.

Requirements

- The SRDF/A device-level write pacing function is supported on Symmetrix arrays running Enginuity 5875 and higher.

Considerations

- Only the device-level pacing values configured for the SRDF group on the R1 side of the SRDF/A session are used.
- In a cascaded SRDF environment:
  - With Enginuity 5876 and lower, device-level write pacing is only supported on the R1->R21 hop of the relationship.
  - With Enginuity 5876 Q4 2012 SR and higher, device-level write pacing is supported on both the R1->R21 and R21->R2 hops of the relationship.
- There is no exemption from device-level pacing as there is for group-level pacing, and the R1 group-level exempt state does not affect device-level pacing.

- In a concurrent SRDF/A environment, device-level pacing is supported on both mirrors of the concurrent R1. In this case, write pacing calculations are performed independently for the two SRDF/A sessions, and the host write I/Os sessions are subject to the greater of the two calculated delays.
- If both group-level pacing and device-level pacing are active for an SRDF/A session, the group-level and device-level delays are calculated independently, and the greater calculated value is used for pacing. Note that as many as four different calculation results may be taken into account for a concurrent R1 device with both mirrors operating in asynchronous mode (group-level pacing for each mirror, device-level pacing for each mirror), using the greatest calculated delay in the calculation.

Setting SRDF/A write pacing attributes

Use the following syntax to set the SRDF/A write pacing attributes for an SRDF group:

```
symrdf -sid <SymmID> -rdfg <GrpNum> [-v] [-symforce]
     [-noprompt] [-i <Interval>] [-c <Count>]

.............

   set rdfa_pace
      <[-dp_autostart <State>]
       [-wp_autostart <State>]
       [-delay <WpaceDelay>]
       [-threshold <WpaceThreshold>]>
       [-both_sides]>
```

To set these group attributes, the remote side must be reachable.

Where:

- `-dp_autostart` (`-dp_aut`) specifies whether SRDF/A device-level pacing is automatically enabled or disabled when an SRDF/A session is activated or deactivated for an SRDF group. Valid state values are on or off (default).
- `-wp_autostart` (`-wp_aut`) specifies whether the SRDF/A group-level pacing feature is automatically enabled or disabled when an SRDF/A session is activated for an SRDF group. Valid state values are on or off (default).
- `-delay` (`-del`) sets the maximum host I/O delay, in microseconds, that the SRDF/A write pacing can cause. Valid values are between 1 and 1000000 microseconds. The default is 50000.
- `-threshold` (`-thr`) sets the minimum percentage of the system write-pending cache at which the Symmetrix array begins pacing host write I/Os for an SRDF group. Valid values are between 1 and 99. The default is 60.
- `-both_sides` sets the SRDF/A write pacing attributes on both the source and target sides of an SRDF/A session. Otherwise, these attributes are only set on the source side.

Note: If you plan on swapping the personalities of the R1 and R2 devices, configure the same SRDF/A write pacing values on both sides.

Examples

The following command automatically activates SRDF/A group-level write pacing for SRDF group 12 with a maximum of a 1000 microsecond delay and a write pending cache threshold of 55 percent (if the calculated delay is less than the specified delay of 1000, then the calculated delay is used):

```
symrdf -sid 134 -rdfg 12 set rdfa_pace -delay 1000 -threshold 55 -wp_autostart on

An RDF Set 'Attributes' operation execution is in
progress for RDF group 12. Please wait...

    SRDF/A Set Pacing Delay (1134,012)............................Started.
    SRDF/A Set Pacing Delay (1134,012)............................Done.
    SRDF/A Set Pacing Threshold (1134,012)........................Started.
    SRDF/A Set Pacing Threshold (1134,012)........................Done.
    SRDF/A Set Group Pacing Autostart (1134,012)..................Started.
    SRDF/A Set Group Pacing Autostart (1134,012)..................Done.
The RDF Set 'Attributes' operation successfully executed
for RDF group 12.
```

When using `-both_sides`, the command output includes two entries for each attribute being applied. One is for the source side and the other is for the target side, as highlighted in the following output:

```
symrdf -sid 432 -rdfg 75 set rdfa_pace -wp_autostart on -delay 500 -threshold 10
   -dp_autostart on -both_sides

An RDF Set 'Attributes' operation execution is in progress for
RDF group 75. Please wait...

    SRDF/A Write Pacing Autostart (0432,075)   ........................Started.
    SRDF/A Write Pacing Autostart (0432,075)   ........................Done.
    SRDF/A Write Pacing Autostart (1134,013)   ........................Started.
    SRDF/A Write Pacing Autostart (1134,013)   ........................Done.
    SRDF/A Write Pacing Delay (0432,075)       ........................Started.
    SRDF/A Write Pacing Delay (0432,075)       ........................Done.
    SRDF/A Write Pacing Delay (1134,013)       ........................Started.
    SRDF/A Write Pacing Delay (1134,013)       ........................Done.
    SRDF/A Write Pacing Threshold (0432,075)   ........................Started.
    SRDF/A Write Pacing Threshold (0432,075)   ........................Done.
    SRDF/A Write Pacing Threshold (1134,013)   ........................Started.
    SRDF/A Write Pacing Threshold (1134,013)   ........................Done.
    SRDF/A Dev Pacing Autostart (0432,075)     ........................Started.
    SRDF/A Dev Pacing Autostart (0432,075)     ........................Done.
    SRDF/A Dev Pacing Autostart (1134,013)     ........................Started.
    SRDF/A Dev Pacing Autostart (1134,013)     ........................Done.

The RDF "Attributes'' operation successfully executed for RDF group 75.
```

You can also use the `symconfigure` command to set SRDF/A write pacing attributes. For more information, see the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide.*

Activating SRDF/A write pacing

You can activate and control group-level and device-level write pacing individually or simultaneously at the group, device group, composite group, and file level. For more information, see "Activate and deactivate SRDF/A write pacing" on page 57.

Identifying devices that cannot be paced in a cascaded SRDF configuration

A source device might not be paced either because it has been set exempt from group-level write pacing or because it is not currently pace-capable.

Source devices (R1 or R21) that are exempt from group-level write pacing are those that have been manually excluded from group-level write pacing via the `-rdfa_wpace_exempt` option of the `symrdf` command. Note that exempt devices can still be paced by device-level write pacing.

The R21 device of an R21›R2 pair is considered not pace-capable if the corresponding R1›R21 SRDF pair is read/write (RW) on the SRDF link and operating in an adaptive copy mode. A device that is not pace-capable cannot be paced by device-level write pacing or group-level write pacing. Solutions Enabler requires the `-force` option for actions that will cause a device to become not pace-capable.

To identify devices that cannot be paced:

1. Use the `symcfg list` command with the `-rdfa` option to determine if the SRDF/A session includes devices that cannot be paced. This command provides the following information related to write pacing:
   - The state of write pacing (group-level and device-level) for the SRDF group
   - Whether write pacing is currently activated and supported
   - Whether write pacing is configured for autostart
   - Whether there are devices in the SRDF/A session that might not be paced either because they have been set exempt from group-level write pacing or because they are not pace-capable.

     For example, to view write pacing information for SRDF group 153, type:

     symcfg list -sid 1134 -rdfg 153 -rdfa

```
Symmetrix ID : 000195701134

                  S Y M M E T R I X   R D F A   G R O U P S

-------- ---------- -------- ----- --- --- --------- ------------------------
                                                        Write Pacing
 RA-Grp  Group      Flags    Cycle Pri Thr Transmit   Delay  Thr GRP DEV FLGS
         Name       CSRM TDA time          Idle Time (usecs) (%) SAU SAU P
-------- ---------- -------- ----- --- --- --------- ------- --- --- --- ----
153 (98) lc153142   .IS- XI.    15  33  50 000:00:00   50000  60 I.- I.- X

Legend:

  RDFA Flags          :
      (C)onsistency   : X = Enabled, . = Disabled, - = N/A
      (S)tatus        : A = Active, I = Inactive, - = N/A
      (R)DFA Mode     : S = Single-session, M = MSC, - = N/A
      (M)sc Cleanup   : C = MSC Cleanup required, - = N/A
      (T)ransmit Idle : X = Enabled, . = Disabled, - = N/A
      (D)SE Status    : A = Active,  I = Inactive, - = N/A
      DSE (A)utostart : X = Enabled, . = Disabled, - = N/A
```

```
  Write Pacing Flags  :
    (GRP) Group-Level Pacing:
      (S)tatus        : A = Active, I = Inactive,  - = N/A
      (A)utostart     : X = Enabled, . = Disabled, - = N/A
      S(U)pported     : X = Supported, . = Not Supported, - = N/A

    (DEV) Device-Level Pacing:
      (S)tatus        : A = Active, I = Inactive,  - = N/A
      (A)utostart     : X = Enabled, . = Disabled, - = N/A
      S(U)pported     : X = Supported, . = Not Supported, - = N/A

    (FLGS) Flags for Group-Level and Device-Level Pacing:
      Devs (P)aceable : X = All devices, . = Not all devices, - = N/A
```

An X in the FLGS P column indicates that all of the devices in the SRDF group can be paced. A period in the FLGS P column indicates that some of the devices in the SRDF group cannot be paced either because they have been set exempt from group-level write pacing or because they are not pace-capable.

2. Use the `symrdf list` command to determine which devices cannot be paced.
   a. Use the `symrdf list` command with the `-rdfa_wpace_exempt` option to identify devices that are exempt from group-level write pacing.
   b. Use the `symrdf list` command with the `-rdfa_not_pace_capable` option to identify devices participating in the SRDF/A session that are not pace-capable.
3. Use the `symdev show` command to obtain additional information about the devices identified in the previous step. This command provides the following information related to write pacing:
   - Whether the device is exempt from group-level write pacing
   - Whether write pacing is currently activated and supported
   - Whether the device is pace-capable

     For example, to view write pacing information for device 00d1, type:

     `symdev show -sid 230 00d1`

```
Device Physical Name     : Not Visible

    Device Symmetrix Name    : 00D1
    Device Serial ID         : N/A
    Symmetrix ID             : 000195700230
      . . .
    RDF Information
        {
        Device Symmetrix Name                  : 00D0
        RDF Type                               : R2
        RDF (RA) Group Number                  : 123 (7A)

        Remote Device Symmetrix Name           : 010B
        Remote Symmetrix ID                    : 000195700046
          . . .
        Device Suspend State                   : N/A
        Device Consistency State               : Disabled
        Device Consistency Exempt State        : Disabled
        RDF R2 Not Ready If Invalid            : Enabled
```

```
        Write Pacing Information
            {
            Pacing Capable                      : Yes
            Configured Group-level Exempt State: Disabled
            Effective Group-level Exempt State : Enabled
            Group-level Pacing State            : Enabled
            Device-level Pacing State           : Disabled
            }

Device RDF State                        : Ready           (RW)
        Remote Device RDF State                 : Write Disabled  (WD)

        RDF Pair State (  R1 <===> R2 )         : Consistent

        Number of R1 Invalid Tracks             : 0
        Number of R2 Invalid Tracks             : 0
```

CHAPTER 5
Concurrent SRDF Operations

This chapter explains how to create and perform operations on concurrent SRDF devices using the SYMCLI. Topics include:

- Overview 142
- Concurrent SRDF operations 144

Overview

In a concurrent SRDF configuration, data on a single source device is remotely mirrored to two target devices at the same time, providing two available copies of data. These mirrors operate independently but concurrently using any combination of SRDF modes.

Concurrent SRDF is valuable for duplicate restarts and disaster recovery, and provides increased flexibility for data mobility and application migrations.

Each of the two concurrent mirrors must belong to a different SRDF group. Figure 15 shows a concurrent SRDF R1 (R11) configuration that uses SRDF group 45 to mirror production data to Symmetrix B and SRDF group 101 to mirror data to Symmetrix C. Copies of the production data are always available at the target sites.

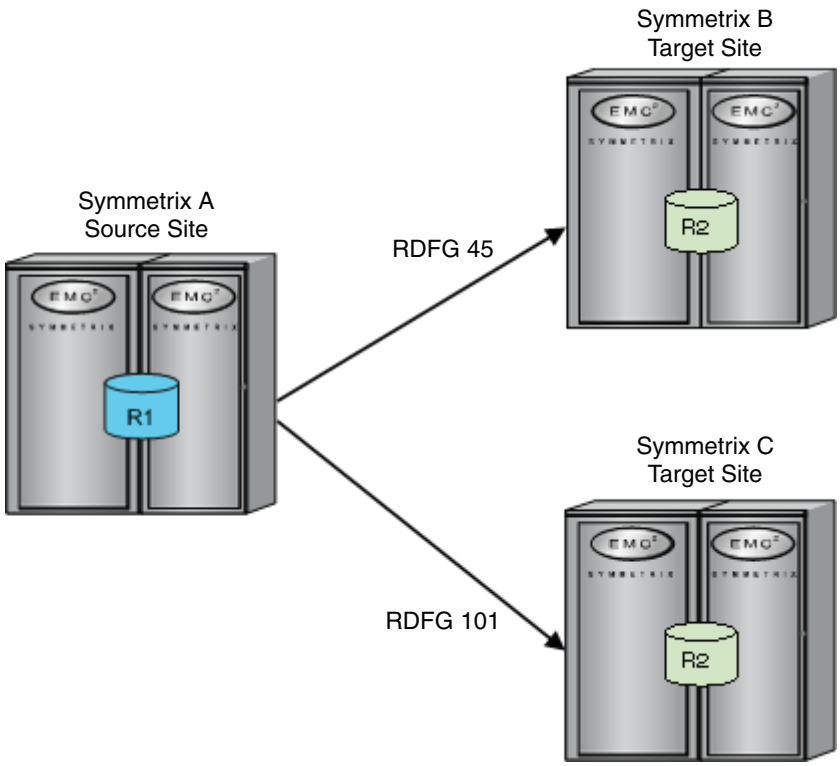


Figure 15 Concurrent SRDF R1 configuration

Using concurrent R2 devices

Concurrent R2 (R22) devices are specifically designed for SRDF/Star configurations to simplify failover situations and improve the resiliency of SRDF/Star applications. In this configuration, an R22 device has two remote mirrors, only one of which can be active (read/write) at a given time. For detailed information on this type of configuration, refer to the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*.

Operating both mirrors in SRDF/S mode

Operating both mirrors in SRDF/S mode ensures that both target sites have exact replicas of the production data. This concurrent configuration requires that all three sites be within your campus or metropolitan area network. Figure 16 shows three sites that are within a limited distance from of one another.

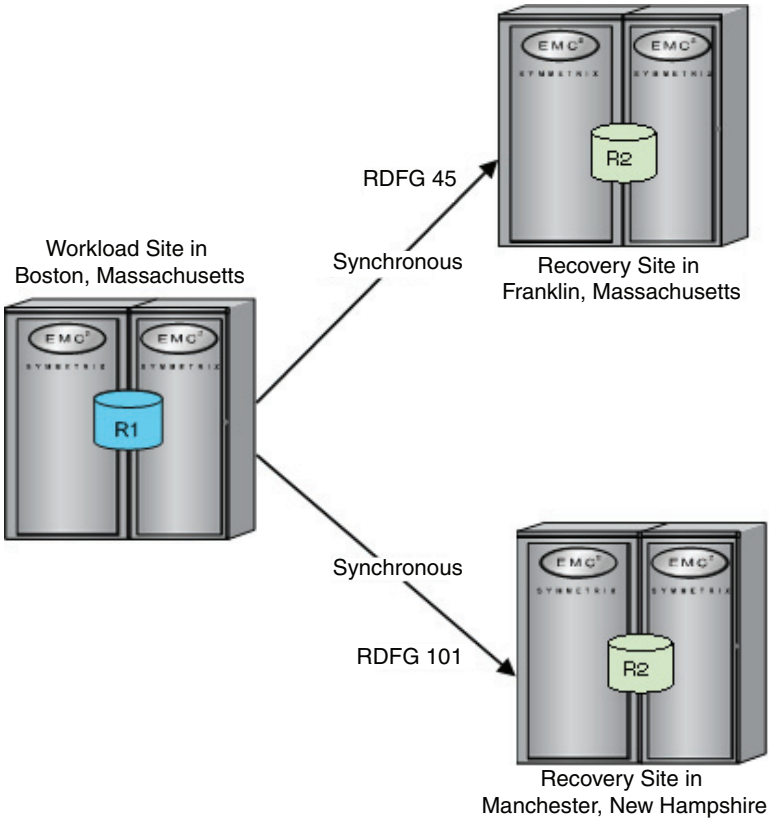


Figure 16 Concurrent SRDF/S mirroring to recovery sites located near the workload site

Operating both mirrors in SRDF/A mode

Prior to the introduction of concurrent SRDF/A, you could synchronously mirror to a recovery site near the production site while asynchronously mirroring to a recovery site far away from the production site. However, if a region-wide disaster occurred near the workload site, you would be at risk of losing both the production data and its synchronous recovery mirror, leaving only the asynchronous recovery mirror intact.

To avoid such a scenario, you can now configure concurrent SRDF/A to asynchronously mirror to recovery sites located at extended distances from the workload site. If the location of the workload site experiences a regional disaster, two copies of the production

data are still available at the remote recovery sites, as shown in Figure 17. In this configuration, the Massachusetts workload site has two recovery sites that are at an extended distance from this site.

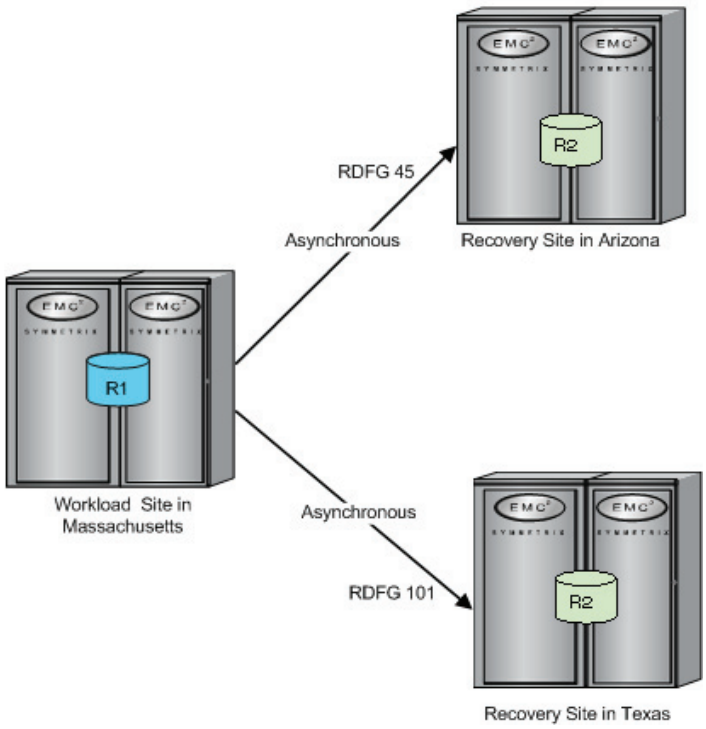


Figure 17 Concurrent SRDF/A mirroring to recovery sites located far away

To simultaneously operate both mirrors in SRDF/A mode, as shown in Figure 17, the concurrent R1 device must reside on a Symmetrix array running Enginuity 5875 or higher. The R2 devices can reside on any Symmetrix array supporting connections to a Symmetrix array running Enginuity 5875 or higher.

Concurrent SRDF operations

With concurrent SRDF, you can build a device group or a composite group containing devices that only belong to the two SRDF groups representing the concurrent remote mirrors. Your device group can also include BCV devices and SRDF devices that are not concurrent SRDF devices but that belong to either one of the concurrent SRDF groups.

Each mirror in a concurrent relationship must belong to a different SRDF group. When controlling or setting concurrent SRDF devices, the `-rdfg` *n* option specifies the SRDF group number (*n*) or remote mirror to control. To perform an operation on both concurrent remote mirrors, use `-rdfg ALL`.

Applicable pair states for concurrent SRDF operations

In a concurrent relationship, there are two separate links, or legs, sending data from an R1 device to two separate R2 devices. You can perform a control operation on one of these legs only if the other leg is in a certain pair state. For more information, refer to "Concurrent SRDF operations and applicable pair states" on page 509.

Establishing concurrent SRDF devices

To create a device group for the concurrent SRDF devices and initially synchronize (establish) the devices across the concurrent SRDF links, follow these steps:

1. Create a concurrent SRDF device group:

   ```
   symdg create ConcGrp -type RDF1
   ```

2. Add all concurrent SRDF devices to the device group:

   ```
   symdg add dev 0001 -g ConcGrp -sid 0001
   symdg add dev 0021 -g ConcGrp
   symdg add dev 002A -g ConcGrp
   ```

3. Establish concurrent SRDF pairs that belong to the device group (first mirror and then second mirror):

   ```
   symrdf -g ConcGrp establish -rdfg 1
   symrdf -g ConcGrp establish -rdfg 2
   ```

   Or, you can use the `-rdfg ALL` option to simultaneously establish both mirrors of each SRDF pair in one command:

   ```
   symrdf -g concGrp -full establish -rdfg ALL
   ```

Note: Business Continuance Volume (BCV) devices cannot contain concurrent SRDF mirrors.

Viewing concurrent SRDF devices

Using the `-concurrent` option with `symrdf list`, you can view all the Symmetrix devices to see which were configured as concurrent SRDF devices:

```
symrdf list [-sid SymmID] -concurrent
```

Using the `-rdfg ALL` option with `symrdf query`, you can view the SRDF states and modes of both remote mirrors of a concurrent SRDF device:

```
symrdf -g DgName query -rdfg ALL
```

Splitting concurrent SRDF devices

You split the concurrent SRDF pair either simultaneously or sequentially. To split the links simultaneously, type:

```
symrdf -g concGrp split -rdfg ALL
```

To split the two remote mirrors one at a time, type:

```
symrdf -g concGrp split -rdfg 1
symrdf -g concGrp split -rdfg 2
```

Enabling consistency protection

You can enable consistency protection for concurrent mirrors using the SRDF name. For more information, refer to "Enabling SRDF consistency protection for concurrent SRDF devices" on page 208 and "Example 5: Consistency protection for concurrent SRDF" on page 396.

Note: With Enginuity 5874 and higher, R1 devices can have two mirrors participating in different consistency groups with SRDF/S consistency protection enabled.

Note: Concurrent R1 devices can have two mirrors participating in different consistency groups with MSC consistency protection enabled. This applies only to Symmetrix arrays running Enginuity 5875 and higher.

Restoring R1 in a concurrent SRDF configuration

You select which target R2 device to use as the mirror for restoring the R1 device. When the `restore` command is executed, both remote mirrors are split and the R1 device is restored from the R2 device in SRDF Group 1, as shown in Figure 18.

```
symrdf -g concGrp restore -rdfg 1
```

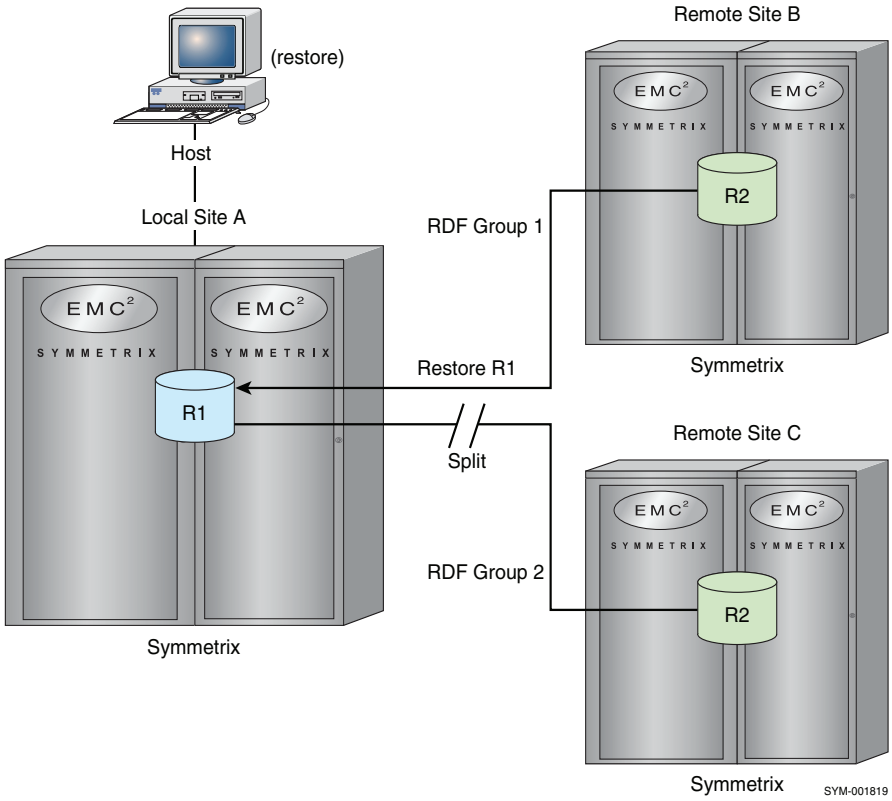


Figure 18 Restoring the source device in a concurrent configuration

After the `restore` operation, the R1 device is synchronized with the R2 mirror belonging to SRDF Group 1, and the R2 device belonging to SRDF Group 2 is still in the split state. You can now re-establish the devices belonging to SRDF Group 2 to a synchronized concurrent SRDF state using the following command:

```
symrdf -g concGrp establish -rdfg 2
```

Restoring both R1 and R2 in a concurrent SRDF configuration

Figure 19 illustrates how to use one of the R2 mirrors to restore both the R1 and the other R2 in a concurrent relationship:

- First, the SRDF Group 2 propagates data from the R2 to the R1, and
- Then the SRDF Group 1 uses this data to restore the other R2 mirror, synchronizing all concurrent SRDF mirrors.

Note: You cannot simultaneously copy data from two remote mirrors to an R1 device.

The following command shows how to restore the R1 using the SRDF Group 2:

```
symrdf -g ConcGrp restore -rdfg 2 -remote
```

The remote data copy option (`-remote`) applies to restore, failback and R1 update control operations.

Note: The `-remote` option can be applied with the `createpair` command for a restore operation to dynamically create a concurrent SRDF pair by adding a second remote mirror. Refer to "Creating dynamic concurrent SRDF pairs" on page 103.

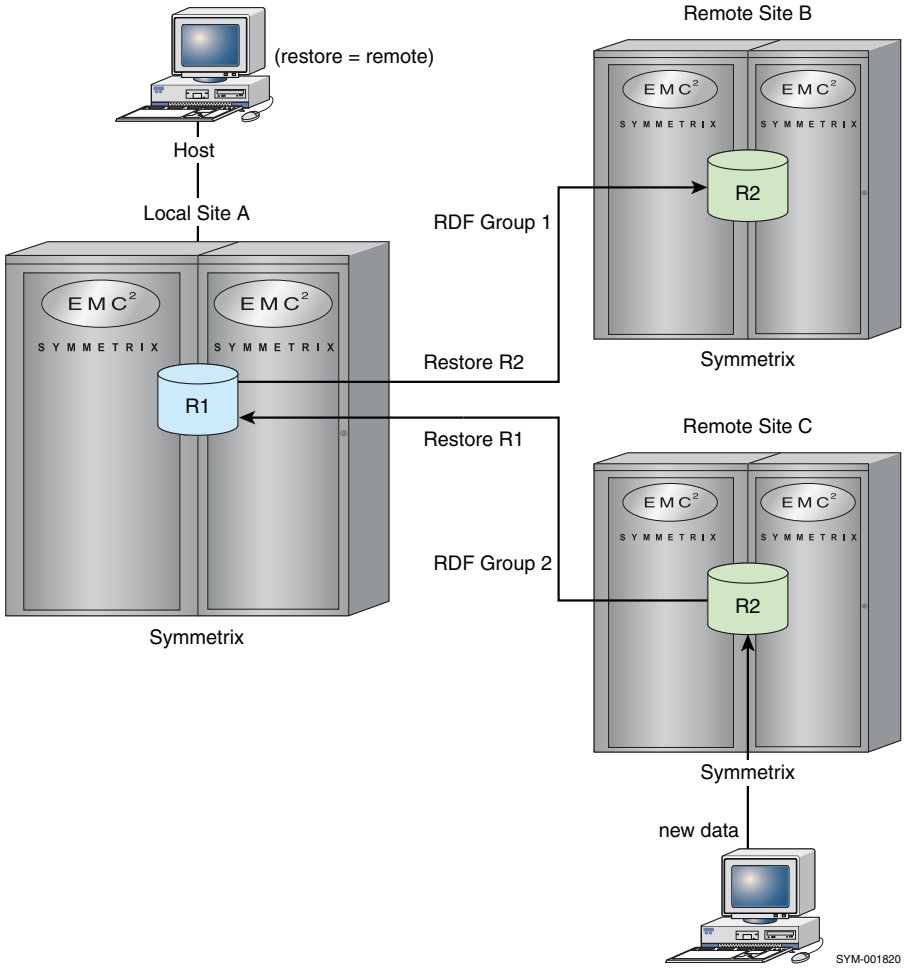


Figure 19 Restoring the source device and mirror in a concurrent SRDF configuration

CHAPTER 6
Cascaded SRDF Operations

This chapter explains how to create and perform operations on devices in a cascaded SRDF configuration using the SYMCLI. Topics include:

- Overview 150
- Setting up cascaded SRDF 152
- SRDF mirror-based controls of an R21 device 155
- SRDF/Extended Distance Protection 161
- Managing a diskless cascaded environment 164

Overview

Cascaded SRDF is a three-way data mirroring and recovery solution that provides enhanced replication capabilities, greater interoperability, and multiple ease-of-use improvements. Cascaded SRDF allows replication between three sites without requiring the need for SRDF BCVs on the second Symmetrix array. A cascaded SRDF configuration does not require three separate site locations, although that is the most common configuration for a disaster recovery solution.

Before Enginuity 5773, an SRDF device could be a source device (R1) or a target device (R2), but could not function in both roles simultaneously. Cascaded SRDF introduces the concept of the dual role R1/R2 device, referred to as an R21 device. The R21 device is both an R1 mirror and an R2 mirror, for use only in cascaded SRDF operations. When thinking of the R21 device, it is easier to understand the concept if you think of it as a mirror type, instead of as a device. The controls for these devices are relationship-based.

The basic cascaded SRDF configuration consists of a primary site (SiteA) replicating data to a secondary site (SiteB) and replicating the same data to a tertiary site (SiteC), as shown in Figure 20.

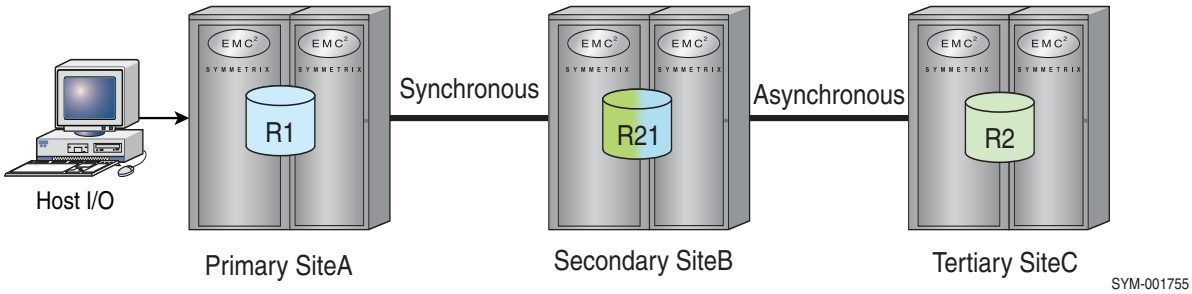


Figure 20 Cascaded SRDF configuration

Note that the Secondary SiteB device is labeled R21. This device is the R2 mirror of the Primary SiteA R1 device, and the R1 mirror of the Tertiary SiteC R2 device. The SiteA and SiteB devices have an SRDF pair state and the SiteB and SiteC devices have an SRDF pair state. These two pair states are separate from each other; however, when performing a control operation on one pair, the state of the other device pair must be known and considered. For a list of control actions and the required SRDF pair states cascaded SRDF configurations, refer to "SRDF Pair State Reference" on page 497.

Note: To perform cascaded SRDF operations with Access Control enabled, you need SRDF BASECTRL, BASE, and BCV access types. For more information, refer to *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

Beginning with Enginuity 5874, the SRDF/Extended Distance Protection (EDP) environment is supported, enabling you to designate an R21 device as a diskless device. The purpose of a diskless R21 device is to directly cascade data to the remote R2 disk device, streamlining the linkage and cost of storage at the middle site. For information about SRDF/EDP, see "SRDF/Extended Distance Protection" on page 161.

Notice that in Figure 20 the data link from SiteA to SiteB is synchronous and the data link from SiteB to SiteC is asynchronous.

While the illustrated modes are the most common, other modes are allowed as shown in Table 12. Asynchronous mode cannot be run on both sides of the link at the same time. Enginuity only supports asynchronous mode on one side in a cascaded configuration.

Table 12 SRDF modes allowed for cascaded configurations

Site A to Site B (R1→R21)	Site B to Site C (R21→R2)
Adaptive copy disk	Asynchronous Adaptive copy disk (non-SRDF/EDP) Adaptive copy write pending (SRDF/EDP only)
Adaptive copy write pending	Asynchronous Adaptive copy disk (non-SRDF/EDP) Adaptive copy write pending (SRDF/EDP only)
Asynchronous (Non-SRDF/EDP)	Adaptive copy disk (non-SRDF/EDP)
Semi-synchronous (for 5671 only)	Asynchronous Adaptive copy disk (non-SRDF/EDP)
Semi-synchronous (for 5773 or higher)	Adaptive copy write pending (SRDF/EDP only)
Synchronous	Asynchronous Adaptive copy disk (non-SRDF/EDP) Adaptive copy write pending (SRDF/EDP only)

Benefits

The main benefit of configuring cascaded SRDF is the capability to continue replicating from the secondary site to the tertiary site if the primary site goes down. This enables a faster recovery at the tertiary site, provided that is where the data operation is restarted.

Cascaded SRDF provides the following benefits and features:

- Sites can be geographically dispersed.
- Faster recovery times at the tertiary site, which is enabled by the capability to continue replicating from the secondary site to the tertiary site if the primary site goes down.
- Zero data loss is achievable up to the point of the primary site failure.

Restrictions

Cascaded SRDF has the following restrictions:

- The secondary site (with the R21 devices) must be on a Symmetrix array running Enginuity 5773 or higher.
- R1 and R2 devices that are paired with R21 devices must be on a Symmetrix array running Enginuity 5671 or Enginuity 5773 and higher.
- An R21 device cannot be paired with another R21 device (R1 -> R21 -> R21 -> R2 is not allowed).
- R21 devices cannot be BCV devices or PPRC devices.
- R21 devices are only supported on GigE and Fibre RAs.

Setting up cascaded SRDF

Cascaded SRDF requires that the R21 device paired with an R2 device be in either asynchronous or adaptive copy disk mode (for non-EDP). The `symrdf` command includes an `-rdf_mode` option with the `createpair` command, which allows you to specify the mode.

```
symrdf -file Filename -sid SymmID -rdfg GrpNum
      [-bypass] [-noprompt] [-i Interval] [-c Count]
      [-v|-noecho] [-force] [-symforce] [-star]

    createpair -type <R1|R2>
      <-invalidate <R1|R2> | -establish | -restore>
      [-rdf_mode <sync|semi|acp_wp|acp_disk|async>]
      [-g NewDg] [-remote]
```

Cascaded SRDF is set up in two SRDF links, which are referred to as hops.

To create the first hop (R1->R21) of the cascaded SRDF device pairs, using the example devices shown in Figure 21, issue the following two commands:

```
symrdf createpair -file TestFile1 -sid 305 -rdfg 210 -type R2 -establish -rdf_mode sync
symrdf createpair -file TestFile2 -sid 321 -rdfg 60 -type R2 -establish -rdf_mode sync
```

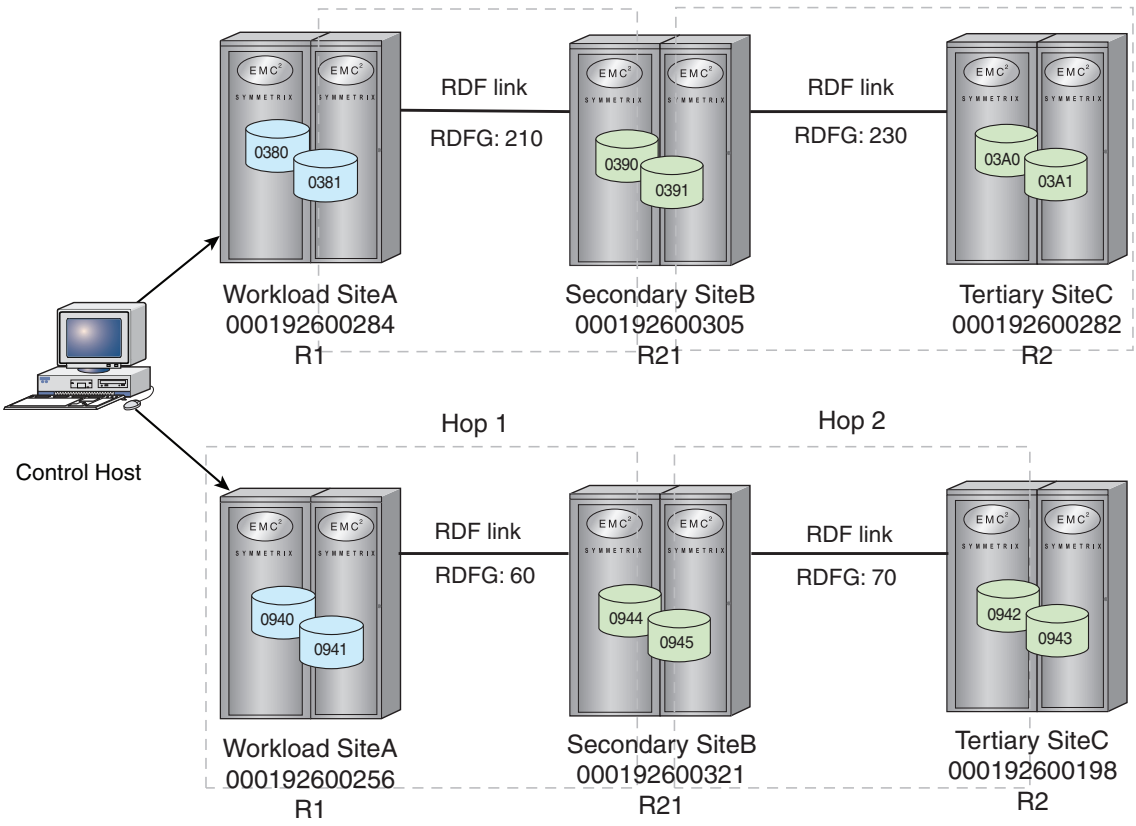


Figure 21 Creating an R21 device

In the previous command examples:

- *TestFile1* has two lines to specify the two device pairs, (0380,0390 on the first line and 0381, 0391 on the second line), and
- *TestFile2* has two lines that specify two device pairs (0940, 0944 on the first line and 0941, 0945 on the second line).

The SRDF device pairs are created, established, and placed in SRDF synchronous mode.

To create the second hop (R21->R2) of the cascaded SRDF device pair, using the example devices shown in Figure 21 on page 152, issue the following two commands:

```
symrdf createpair -file TestFile3 -sid 305 -rdfg 230 -type R1 -establish -rdf_mode acp_disk
symrdf createpair -file TestFile4 -sid 321 -rdfg 70 -type R1 -establish -rdf_mode acp_disk
```

In this example command:

- *TestFile3* has two lines to specify the two device pairs, (0390, 03A0 on the first line and 0391, 03A1 on the second line), and
- *TestFile4* contains two lines to specify two device pairs (0944, 0942 on the first line and 0945, 0943 on the second line).

The SRDF device `0390` is already the R2 device in the R1 ->R21 device pair (0380,0390) and will be the R1 device in the R21-> R2 relationship in the second hop. The SRDF mode is set to `acp_disk` because the SRDF mode on the first hop is synchronous.

Applicable pair states for cascaded SRDF operations

In a cascaded relationship, control operations are only allowed for the pair R1`test_group_cg`R21 when the R21->R2 pair is in a specific pair state. For more information, refer to "Cascaded SRDF operations and applicable pair states" on page 502.

Cascaded SRDF rules

The following rules apply when creating SRDF devices using the `-rdf_mode` option:

- If the first device added to an SRDF group is in asynchronous mode (`-rdf_mode async)`, all subsequent devices added to the SRDF group must also be added in asynchronous mode.
- If you do not specify a mode:
  - For Solutions Enabler versions lower than 7.4, the default is synchronous.
  - For Solutions Enabler version 7.4 and higher, the option file setting SYMAPI_DEFAULT_RDF_MODE is used (which defaults to adaptive copy).
- When adding the first device to an SRDF group with `-rdf_mode sync|semi|acp_wp|acp_disk`, subsequent devices cannot be added in asynchronous mode (`-rdf_mode async`).
- Enginuity 5874.228.182 and higher supports cascaded SRDF operations with thin devices. A mixture of thin and (non-diskless) thick devices is supported with the following restrictions:
  - The Symmetrix arrays that contain the thin devices must be running Enginuity 5876 Q42012 SR and higher.

- For FBA devices, the Symmetrix arrays that contain the thick devices must be running either Enginuity 5773.50154 or Enginuity 5876 Q42012 SR and higher.
- For CKD devices, the Symmetrix arrays that contain the thick devices must be running either Enginuity 5773.50154, Enginuity 5875.64025, or Enginuity 5876 Q42012 SR and higher.

The following rules apply when creating R21 SRDF pairs:

- Solutions Enabler blocks the creation of (R1 -> R21) when the device, which will be the R21 device, is currently an R1 device and is in synchronous or adaptive copy write pending mode (non diskless only). For diskless devices, the Solutions Enabler blocks the creation of an R1 device when it is operating in adaptive copy disk.
- Solutions Enabler blocks the creation of an R21 device if both SRDF groups for the R21 device are not on a Fibre or GigE director.
- Solutions Enabler enforces that the same SRDF group cannot be configured for both R21 device mirrors.

Cascaded SRDF devices can also be created using the `symconfigure add mirror` command. Refer to the *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide* for information about using the `symconfigure` command.

RDF21 SRDF groups

Device groups and composite groups can be created to contain R21 devices as standards. These groups can be identified with an SRDF group type: RDF21. The following syntax examples show how to create an RDF21 device group called `test_group_dg` and a composite group called `test_group_cg`:

```
symdg -type RDF21 create test_group_dg
symcg -type RDF21 create test_group_cg
```

To create an RDF1 composite group called `testcg`, add devices and set an SRDF group name to `name1`, use the following example:

```
symcg -type rdf1 create testcg
symcg -cg testcg addall dev -sid 284 -rdfg 210
symcg -cg testcg addall dev -sid 256 -rdfg  60
symcg -cg testcg set -name name1 -rdfg 284:210,256:60
```

Listing cascaded SRDF devices

The `symrdf list` command with the `-R21` option displays all R21 devices. This option cannot be specified in the same command with the `-R1` or `-R2` option.

The `-cascade` option will list all R21 devices and the R1 and R2 devices with which they are paired. It will also list R1 and R2 devices that are participating in cascaded SRDF relationships. Using the `-cascade` option in conjunction with either the `-R1`, `-R2`, or `-R21` option displays only R1, R2, or R21 devices that are participating in a cascaded SRDF relationship.

Because R21 devices and the devices with which they are paired are considered concurrent SRDF devices, using the `-concurrent` flag also displays these devices.

SRDF mirror-based controls of an R21 device

When querying or controlling an R1 device that is participating in a cascaded SRDF relationship, the terms first hop and second hop will be used for the R1-› R21 device pair and the R21-›R2 device pair respectively. This is also true when controlling an R2 device that is participating in a cascaded SRDF relationship, but the first hop will represent the R2-›R21 relationship and the second hop will represent the R21-› R1 relationship.

When performing controls against one pair relationship, the state of the other pair relationship determines whether the operation will be allowed. The SRDF state of the R21 device in a cascaded relationship is determined as follows:

- The SRDF pair state of the R1 -› R21 device is determined by the RA status.
- The SRDF pair state of the R21 -› R2 mirror is determined by the SA status.

Figure 22 illustrates how the R21 SRDF device state is determined and how each SRDF mirrored pair state is determined.

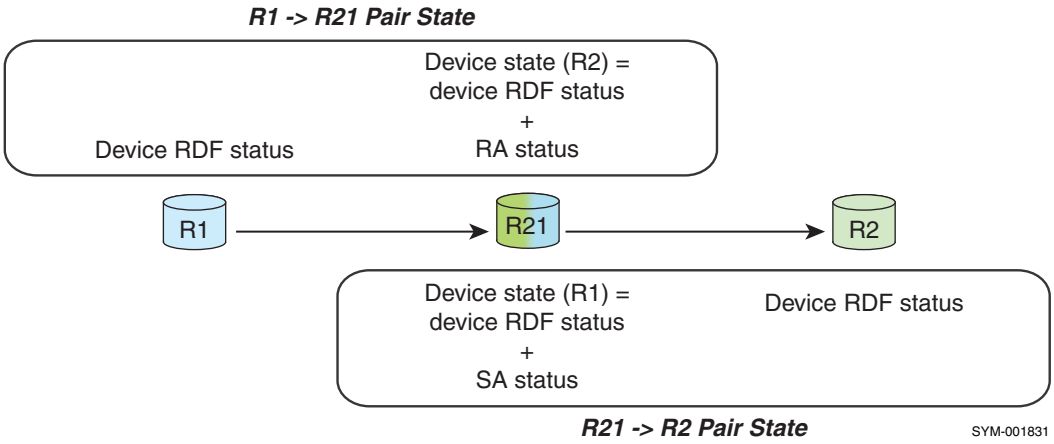


Figure 22 Determining cascaded SRDF pair state

Device actions, such as `rw_disable r1` and `rw_enable r1`, only modify the SA status of the R21 device. For example, if a `rw_enable r1` is performed against the R1 -› R21 pair, and the R21 has a device SA status of `WD`, the overall device SRDF state will remain `WD`. To make an R21 device `rw_enable` to the host, the status of both the R1 mirror and the status of the R2 mirror must be modified. In other words, both a `rw_enable r1` against the R21 -› R2 pair and a `rw_enable r2` against the R1 -› R21 pair must be performed, to accomplish this.

Note: If either the R1 or the R2 mirror of an R21 SRDF device is made `NR` or `WD`, the R21 device will be `NR` or `WD` to the host.

Refer to "Cascaded SRDF operations and applicable pair states" on page 502 for a list of the acceptable pair states for cascaded SRDF operations.

Hop 2 controls

Controls are allowed from hosts connected to either the Symmetrix containing the R1 device, the Symmetrix containing the R21 device, or the Symmetrix containing the R2 device.

You can use the `-hop2` option to control an SRDF device that is two hops away. In order to do this, its concurrent SRDF device on the secondary site (one hop away) must be an R11, R21, or R22 device.

The `-hop2` option allows the SRDF `set` operation to target the group's second-hop devices in a cascaded relationship. For example, given an RDF1 group, the action targets the R21->R2 pair of the R1->R21->R2 relationship. The `-hop2` option can be used with device groups and composite groups. It works on STDs (-hop2) and local BCVs (`-bcv -hop2`) for the following:

- R21->R2 relationship for an RDF1 device group or composite group
- R1->R21 relationship for an RDF2 device group or composite group

Figure 23 shows that the location of the hop-2 devices is dependent on the location of the controlling host. In the top example of Figure 23, the controlling host is the Workload SiteA, therefore a control operation using a command with the `-hop2` option will act on the device pair in the Symmetrix array from SiteB to SiteC.

Conversely, when the controlling host is attached to SiteC, as shown on the bottom of Figure 23, a control operation using a command with the `-hop2` option acts on the device pair in the Symmetrix from SiteB to Site A.

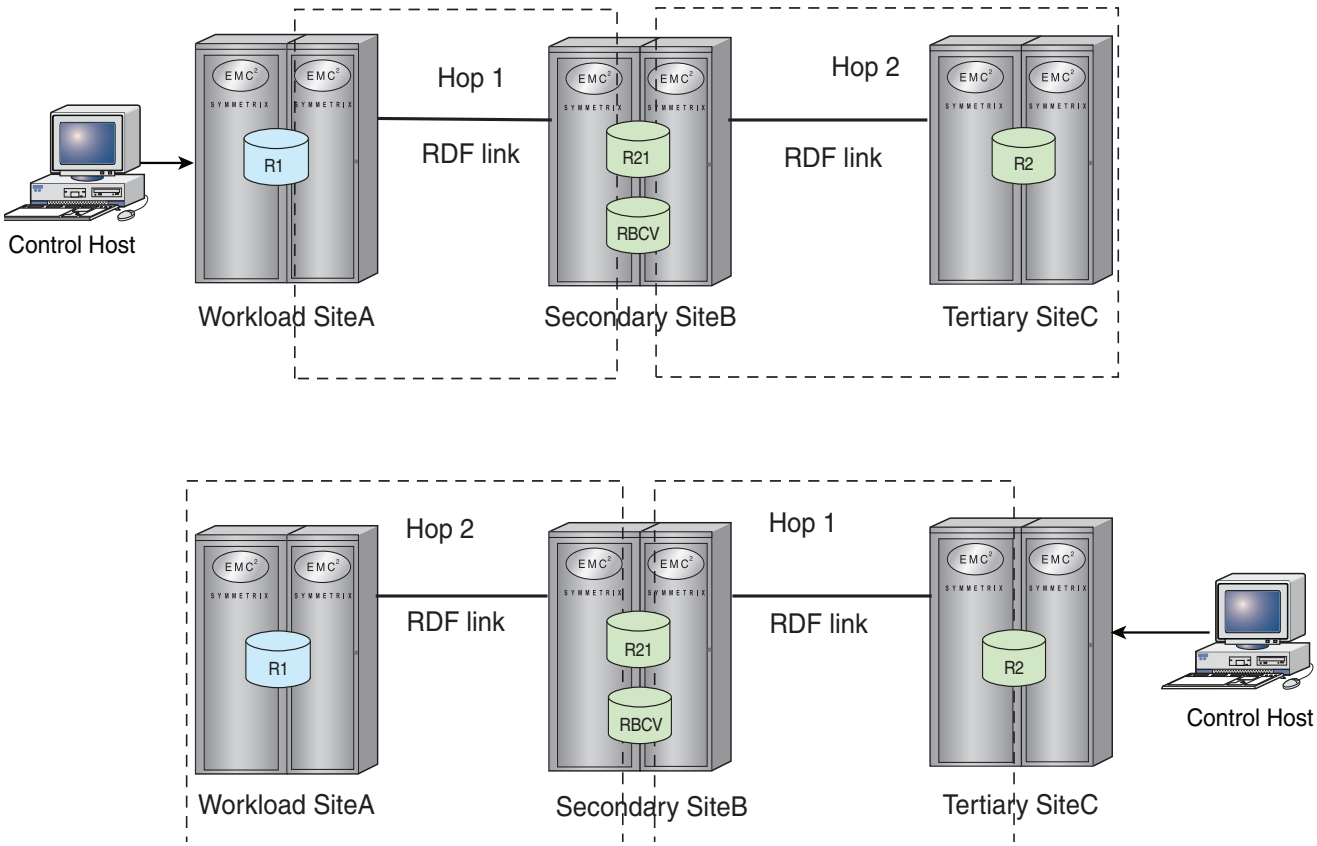


Figure 23 Location of hop-2 devices

Examples of hop 2 controls

Below are examples of controls using the `-hop2` option for SRDF controls:

1. For composite groups, the `-hop2` parameter can be used with `-rdfg name:` to indicate the control action should operate on the second hop SRDF relationship for the specified `-rdfg name:` for all devices in a Cascading SRDF relationship.

   For example, assume we have the following composite group with 4 devices spread across two Symmetrix arrays:

   ```
   CG: testcg     cg type: RDF1   with  R1->R21->R2

   Sym: 000192600284  / rdf group  210   /  rdfg name:  name1
   R1 device  0380
   R1 device  0381


   Sym: 000192600256  / rdf group   60   /  rdfg name:  name1
   R1 device  0940
   R1 device  0941
   ```

2. The following command only operates on the R21->R2 SRDF relationships associated with all the R1 devices using SRDF groups named `name1`:

   `symrdf -cg testcg -rdfg name:name1 -hop2 establish`

Querying hop 2 information

To display information about the second hop SRDF pair of a cascaded SRDF relationship, use the `-hop2` option with the `symrdf query` command. The following output is an example of a `symrdf query testcg -hop2` command.

`symrdf -cg testcg -rdfg name:name1 -hop2 query`

```
Composite Group Name      :  testcg
Composite Group Type      :  RDF1
Number of Symmetrix Units :  2
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192600284    (Microcode Version: 5874)
Hop-2 Symmetrix ID                : 000192600305    (Microcode Version: 5874)
Hop-2 Remote Symmetrix ID         : 000192600282    (Microcode Version: 5874)
RDF (RA) Group Number             : 210 (D1)
Hop-2 RDF (RA) Group Number       : 230 (E5)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ------------------------ ----- ------ ------------
DEV001   0390 RW       0       0 RW 03A0 WD       0       0 C.D.  .  -   Synchronized
DEV002   0391 RW       0       0 RW 03A1 WD       0       0 C.D.  .  -   Synchronized

Symmetrix ID                      : 000192600256    (Microcode Version: 5874)
Hop-2 Symmetrix ID                : 000192600321    (Microcode Version: 5874)
Hop-2 Remote Symmetrix ID         : 000192600198    (Microcode Version: 5874)
RDF (RA) Group Number             :  60 (3B)
Hop-2 RDF (RA) Group Number       :  70 (45)
```

```
        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------- ----- ------ ------------
               ST                 LI      ST                       C  S
  Standard     A                   N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv   K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks    S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV003   0944 RW       0       0 RW 0942 WD       0       0 C.D.  .  -   Synchronized
DEV004   0945 RW       0       0 RW 0943 WD       0       0 C.D.  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Some items to note in the previous query output:

- Number of SRDF (RA) Groups — Represents the number of R1 -› R21 SRDF groups in the composite group.
- Symmetrix ID — Represents the Symmetrix ID of the R1 device.
- Hop-2 Symmetrix ID — Represents the Symmetrix ID of the R21 device.
- Hop-2 Remote Symmetrix ID — Represents the Symmetrix ID of the R2 device.
- SRDF (RA) Group Number — Represents the SRDF group of the R1 device.
- Hop-2 SRDF (RA) Group Number — Represents the SRDF group of the R21 device.
- Total — Sums the invalid tracks (and MB) across all displayed R21 -› R2 SRDF groups (that is, it sums all hop-2 invalid tracks).

Note: With an R1-›R21-› R2 configuration, issuing a `query -hop2` from an RDF1 composite group indicates that the query should show the relationship of the R21-› R2 device pairs. Thus the query displays the R21 device from the R1 mirror point of view (and vice versa for RDF2 CG).

To see both hops of the RDF1 or RDF2 CG that contains devices in a cascaded SRDF relationship, use the `symrdf -cg query` command with the `-hop2` and the `-detail` options. Below is an example of the output of this command when applied to the same CG (`testcg`) used earlier to illustrate the `-hop2 query` output. The format of the output has changed to associate the cascaded pair with the appropriate local pair.

Note: The `-detail` option is not supported for a device group.

To display detailed information about the second hop SRDF pair of a cascaded SRDF relationship, use the `-detail` option with the `symrdf query` command. The following output is an example of a `symrdf query -hop2` command:

symrdf query -cg testcg -rdfg name:name1 -hop2 -detail

```
Composite Group Name      :  testcg
Composite Group Type      :  RDF1
Number of Symmetrix Units :  2
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names:
    {
    RDFG Name                          : name1
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000192600284    (Microcode Version: 5874)
Remote Symmetrix ID               : 000192600305    (Microcode Version: 5874)
RDF (RA) Group Number             : 210 (D1) - name1

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ------------------------- ----- ------------
DEV001   0380 RW       0       0 RW 0390 WD       0       0 S.... Synchronized
DEV002   0381 RW       0       0 RW 0391 WD       0       0 S.... Synchronized

Hop-2
    {
    Symmetrix ID                      : 000192600305    (Microcode Version: 5874)
    Remote Symmetrix ID               : 000192600282    (Microcode Version: 5874)
    RDF (RA) Group Number             : 230 (E5)

            Source (R1) View             Target (R2) View       MODES
    --------------------------------  ------------------------- ----- ------------
                  ST                 LI      ST
      Standard     A                  N       A
    Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
    Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
    -------------------------------- -- ------------------------- ----- ------------
    DEV001   0390 RW       0       0 RW 03A0 WD       0       0 C.D.. Synchronized
    DEV002   0391 RW       0       0 RW 03A1 WD       0       0 C.D.. Synchronized
    }

Symmetrix ID                      : 000192600256    (Microcode Version: 5874)
Remote Symmetrix ID               : 000192600321    (Microcode Version: 5874)
RDF (RA) Group Number             :  60 (3B) - name1

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ------------------------- ----- ------------
DEV003   0940 RW       0       0 RW 0944 WD       0       0 S.... Synchronized
DEV004   0941 RW       0       0 RW 0945 WD       0       0 S.... Synchronized

Hop-2
{
    Symmetrix ID                      : 000192600321    (Microcode Version: 5874)
    Remote Symmetrix ID               : 000192600198    (Microcode Version: 5874)
    RDF (RA) Group Number             :  70 (45)

            Source (R1) View             Target (R2) View       MODES
```

```
    -------------------------------  ------------------------- ----- ------------
                  ST                 LI      ST
      Standard     A                  N       A
    Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
    Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
    ------------------------------- -- ----------------------- ----- ------------
    DEV003   0944 RW       0       0 RW 0942 WD       0       0 C.D.. Synchronized
    DEV004   0945 RW       0       0 RW 0943 WD       0       0 C.D.. Synchronized
    }

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

  Hop-2 Track(s)       0       0                  0       0
  Hop-2 MBs          0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

Some items to note in the previous query output:

- Symmetrix ID — Represents the Symmetrix ID of the R1 device if outside a Hop-2 {. . .} group, or the Symmetrix ID of the R21 device if inside a Hop-2 {. . .} group.
- Remote Symmetrix ID — Represents the Symmetrix ID of the R21 device if outside a Hop-2 {. . .} group, or the Symmetrix ID of the R2 device if inside a Hop-2 {. . .} group; had this been an RDF2 CG, then Remote Symmetrix ID inside a Hop-2 {. . .} group would represent the Symmetrix ID of the R1 device.
- SRDF (RA) Group Number — Represents the SRDF group from the R1->R21 devices if outside a Hop-2 {. . .} group, or the SRDF group from the R21->R2 devices if inside a Hop-2 {. . .} group; had this been an RDF2 CG, then SRDF (RA) Group Number inside a Hop-2 {. . .} group would represent the SRDF group from the R21->R1 devices.

Note: As with the `-hop2 query` of the same CG illustrated earlier, different R21->R2 SRDF groups are reported separately.

Listing R21 devices

The `symrdf list` command includes the options `-R21` and `-cascade` for displaying cascaded SRDF devices. Use the `-R21` option to display only the R21 devices. Use the `-cascade` option to display all devices (R1, R21, R2) configured in the cascaded SRDF relationship.

The `symrdf list` output also returns the SRDF Mirror Type associated with the SRDF group. When viewing the output for a R21 device, it can be determined if the R1 or R2 mirror information is being displayed.

The following output example shows the Mirror Type in bold text.

```
symrdf list -sid 305 -cascaded

Symmetrix ID: 000192600305
```

```
                              Local Device View
-----------------------------------------------------------------------
                     STATUS     MODES                       RDF  S T A T E S
Sym        RDF      ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G    SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------  ---------  ----- -------  ------- --- ---- -------------

0390 0380  R21:210  RW WD RW   S..2.       0        0 WD  RW   Synchronized
     03A0  R21:230  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0391 0381  R21:210  RW WD RW   S..2.       0        0 WD  RW   Synchronized
     03A1  R21:230  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0920 0920   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0921 0921   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0922 0922   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0923 0923   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0924 0924   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0925 0925   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0926 0926   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended
0927 0927   R2:35   ?? NR NR   C.D2.       0        0 NR  RW   Suspended

Total                                -------- --------
  Track(s)                                  0        0
  MB(s)                                   0.0      0.0


Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Mirror) T(ype)       : 1 = R1, 2 = R2
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

SRDF/Extended Distance Protection

SRDF/Extended Distance Protection (EDP) streamlines a cascaded SRDF linkage out to the remote site with a direct (diskless) connection. In a cascaded SRDF/EDP environment, the cascaded R21 devices are designated as diskless devices. This provides replication between the source (site A) and remote target (site C) without requiring disks at the middle Symmetrix site B, as shown in Figure 24.

Note: SRDF/EDP is available with Symmetrix arrays running Enginuity 5874 or higher.

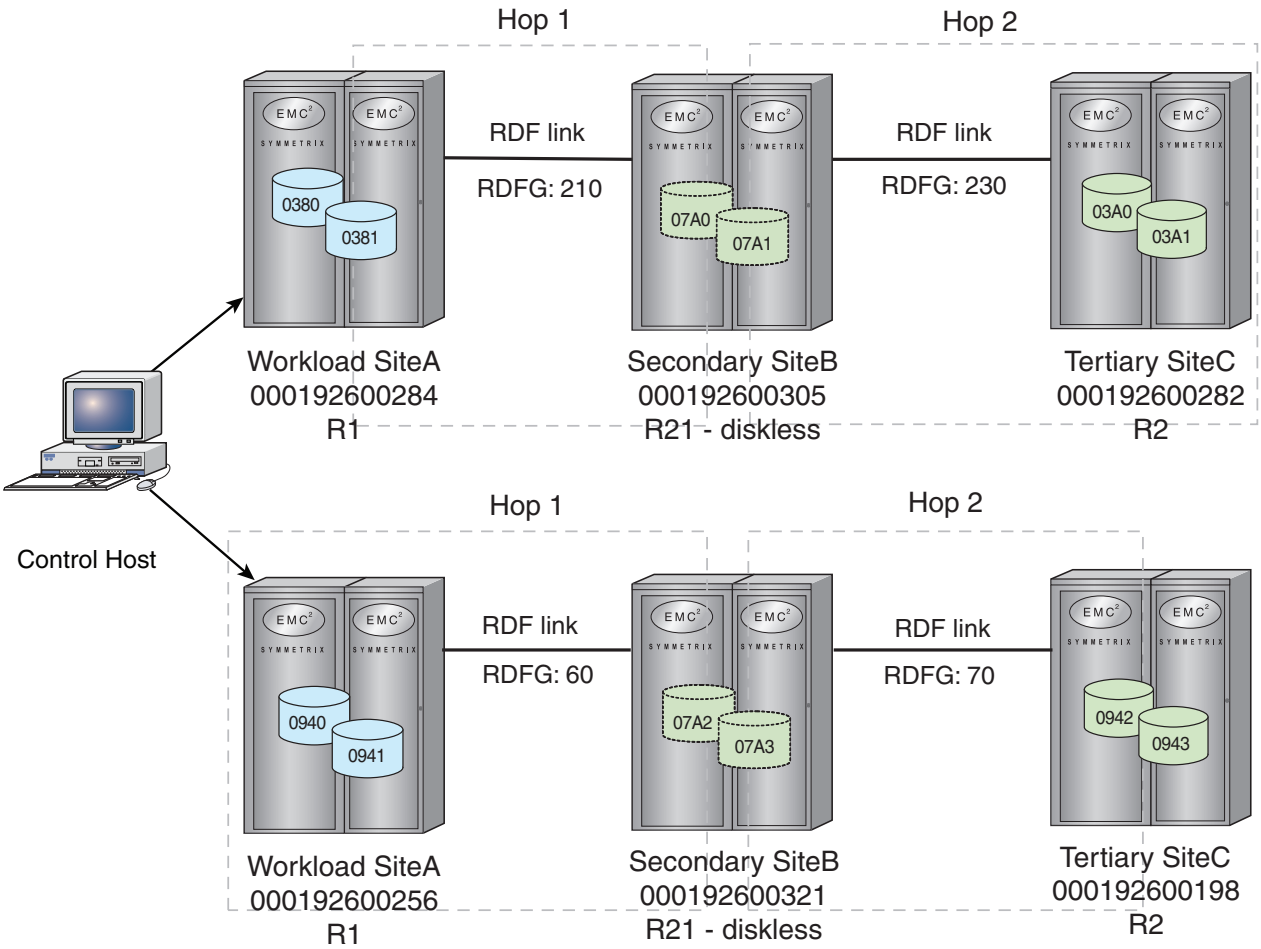


Figure 24 Creating an R21 diskless device

In comparison, a standard cascaded SRDF device in a Symmetrix array at the middle site (B), referred to as an R21 device, assumes a dual role of both an R1 and an R2 simultaneously in a mirror relationship to the local site (A) and the remote site (C). In this standard three site cascaded environment, a regular R21 disk device has its own local mirrors so there are three full copies of data, one at each of the three sites. However with EDP, the diskless R21 device at the middle site has no local mirrors. As it is in the middle Symmetrix at site B of the total SRDF link to the remote target site, there are only two full copies of data, one on the source R1 disk device and one on the remote target R2 disk at site C device.

The introduction of a diskless cascaded SRDF linkage means that the R21 device is a new type of I/O connection device which does not have any local mirrors. Since the device has no local disk space allocated to store the user data, it reduces the cost of having disk storage in the R21 Symmetrix array.

The purpose of a diskless R21 device is to directly cascade data to the remote R2 disk device, streamlining the linkage. When using a diskless R21 device, the changed tracks received from the R1 mirror are saved in cache until these tracks are sent to the R2 disk device. Once the data is sent to the R2 device and the receipt is acknowledged, the cache slot is freed and the data no longer exists on the R21 Symmetrix.

Note: Diskless SRDF is supported in conjunction with the SRDF/A Delta Set Extension feature (see "Activate and deactivate SRDF/A Delta Set Extension" on page 56).

The SRDF/Extended Distance Protection feature requires Symmetrix Enginuity 5874 or higher. However, the Enginuity level for the Symmetrix array(s) containing the SRDF partner(s) of the diskless SRDF devices at the local site (R1) and the remote site (R2) can be running Enginuity 5773 or higher.

SRDF/EDP rules

The following rules apply when creating diskless SRDF devices:

- For Enginuity 5773, a patch is required to connect to a diskless device.
- A diskless device cannot be mapped to the host. Therefore, no host is able to directly access a diskless device for I/O data (read or write).
- The diskless SRDF devices are only supported on GigE and Fibre RAs.
- Other Symmetrix replication technologies (TimeFinder/Snap, TimeFinder/Clone, Open Replicator, and Federated Live Migration) do not work with diskless devices as the source or the target of the operation.
- The `symreplicate` command returns an error if a diskless device is found in the configuration.
- Diskless devices are not supported with thin CKD devices.
- The R1 and R2 volumes must be *both* thin or *both* standard. For example:
  - Thin R1-› diskless R21-›thin R2, *or*
  - Standard, fully provisioned R1 -› diskless R21 -› standard, fully provisioned R2.

Setting up cascaded SRDF/EDP

Cascaded SRDF/EDP requires that the R21 device paired with an R2 device be in asynchronous or adaptive copy write pending mode. The `symrdf` command includes an `-rdf_mode` option with the `createpair` command, which allows you to specify the mode.

```
symrdf -file Filename -sid SymmID -rdfg GrpNum
      [-bypass] [-noprompt] [-i Interval] [-c Count]
      [-v|-noecho] [-force] [-symforce] [-star]

    createpair -type <R1|R2>
      <-invalidate <R1|R2> | -establish | -restore>
      [-rdf_mode <sync|semi|acp_wp|acp_disk|async>]
      [-g NewDg] [-remote]
```

Cascaded SRDF/EDP is set up in two SRDF links (also called hops). To create the first hop (R1-›R21) of a cascaded SRDF/EDP device pairs, using the example devices shown in Figure 24 on page 162, issue the following two commands:

```
symrdf createpair -file TestFile1 -sid 305 -rdfg 210 -type R2 -invalidate R2 -rdf_mode sync
symrdf createpair -file TestFile2 -sid 321 -rdfg 60 -type R2 -invalidate R2 -rdf_mode sync
```

In this example command:

- *TestFile1* contains four devices (`0380,0381,07A0,07A1`) and
- *TestFile2* contains four devices (`0940,0941,07A2,07A3`).

The SRDF device pairs are created and placed in synchronous mode.

With SRDF/EDP, you cannot bring devices Read Write on the link until the diskless devices are designated as being R21s. Here the `-invalidate R2` option is used instead of the `-establish` option. The `-invalidate R2` option requires that the R2 devices be not ready or Write Disabled. However, since the R21 devices in this example are diskless and cannot be mapped, the write disable step is not required.

To create the second hop (R21->R2) of the cascaded EDP device pair, using the example devices shown in Figure 24 on page 162, issue the following two commands:

```
symrdf createpair -file TestFile3 -sid 305 -rdfg 230 -type R1 -establish -rdf_mode acp_wp
symrdf createpair -file TestFile4 -sid 321 -rdfg 70 -type R1 -establish -rdf_mode acp_wp
```

In this command example:

- *TestFile3* contains four devices (07A0,07A1,03A0, 03A1) and
- *TestFile4* contains four devices (07A2,07A3,0942,0943).

The SRDF device 07A0 is already the R2 device in the R1-> R21 device pair (0380,07A0) and will be the R1 device in the R21->R2 relationship in the second hop. The SRDF mode is set to acp_wp because the SRDF mode on the first hop is synchronous.

At the end of this createpair step, the diskless devices are now R21 devices. Since the rule for SRDF/EDP requires that the R21->R2 devices be Read Write on the link, the `-establish` option is used with the `createpair` command.

To make the R1 device pairs (read write) in the R1->R21 hop on the link, issue the following two commands:

```
symrdf establish -file TestFile1 -sid 305 -rdfg 210
symrdf establish -file TestFile2 -sid 321 -rdfg 60
```

Managing a diskless cascaded environment

Diskless devices perform only as R21 devices in a cascaded environment. Configuring a diskless R1, R2, or R22 device should only be used as an intermediate stage of creating a diskless R21 device.

Since a diskless SRDF device does not have local mirrors, fewer controls are supported. The following operations are blocked when controlling a single R1->R2 relationship that is not part of a cascaded SRDF Relationship (in other words, R1->R2, R2<-->R2, or R1->R22<-R1), or is not going to become part of a cascaded SRDF relationship (for the `createpair` action):

- Establish, resume, restore, failback, R1_update, merge
- Failover if the R2 is a diskless device
- Createpair –restore or –establish

- Refresh R1 or swap –refresh R1
- Refresh R2 or swap –refresh R2
- Ready/not_ready R1 of a diskless R1 device
- Ready/not_ready R2 of a diskless R2 device

A diskless SRDF device may not be paired with another diskless SRDF device.

When setting devices configured in a SRDF group into asynchronous mode, all the devices in the SRDF group must be either diskless or non-diskless.

When the R21->R2 hop is in adaptive copy WP mode, the system blocks the setting of any skew limit. The adaptive copy WP mode behaves as if the skew is infinite.

The link between R21->R2 must be ready (RW) before making the R1->R21 link ready (RW). If not, Enginuity drops the diskless R1->R21 devices NR on the link when the R21->R2 state is NR on the link.

SRDF control and set operations

This section addresses control actions in a diskless environment, typically using the `symrdf` command.

SRDF control and set operations for diskless environments can be targeted to composite and device groups, and files that contain both diskless and non-diskless devices. The processing for SRDF pairs containing diskless devices is no different than the processing for SRDF pairs containing non-diskless devices, although some new restrictions apply when you are controlling a cascaded SRDF environment that includes diskless devices.

Note that you can control SRDF pairs with diskless devices and without diskless devices in a single control operation if some of the R21 devices in the group are diskless and others are not. In addition to the support for all the existing configurations, the following configurations are supported when the R21 is a diskless SRDF device:

R1->R21->R2

R11->R21->R2

R11->R21->R22

Setting the mode to asynchronous against an SRDF group containing diskless and non-diskless devices is not allowed. Table 12 on page 151 lists the modes allowed for cascaded SRDF configuration.

With an R1->R21->R2 configuration, where the R21 is diskless, only the combinations shown in Table 13 are allowed. All other combinations are blocked. If synchronous mode (the default) is not allowed, make sure you specify a valid SRDF mode when creating these device pairs.

Table 13 SRDF modes allowed for cascaded SRDF/EDP (page 1 of 2)

R1→R2	R21→R2
Synchronous Adaptive copy disk Adaptive copy write pending	Asynchronous

Table 13 SRDF modes allowed for cascaded SRDF/EDP (page 2 of 2)

R1→R2	R21→R2
Synchronous Adaptive copy disk Adaptive copy write pending	Adaptive copy write pending

Note: The Adaptive copy write pending and Asynchronous combination in Table 13 is incapable of reaching a Consistent state. The R21->R2 hop hangs in the SyncInProg state with 0 invalid tracks. To reach a Consistent state, switch Adaptive Copy - WP to Asynchronous.

Dynamic SRDF controls

The dynamic SRDF controls (`createpair`, `deletepair`, `swap_personality`, `movepair`, and `failover -establish` actions) can be used to create and manage diskless SRDF device relationships.

The following rules apply for these operations:

- A diskless SRDF device can only be configured on a Fibre or GigE SRDF director.
- A `createpair` action is blocked when both sides are diskless devices.
- The `createpair` and `movepair` actions are blocked if the action results in a mixture of diskless and non-diskless devices in an SRDF group containing devices in asynchronous mode.
- The `createpair`, `movepair`, `swap_personality`, and `failover -establish` actions will be blocked if the action will result in a violation of the allowable SRDF modes as outlined in "SRDF control and set operations" on page 165.
- The `createpair` action is blocked if the action results in an R1->R21->R2 relationship where the R1 and the R2 are the diskless devices.

SRDF query support

Since a diskless SRDF device has no local mirrors, no local invalid tracks are ever reported for the device. This means that a diskless R1 device, when queried, will never show any R1 invalid tracks. A diskless R2 device will never show any R2 invalid tracks. A diskless R21 will not show any R1 invalid tracks when querying it from the R21->R2 relationship point of view and will not show any R2 invalid tracks when queried from the R1->R21 relationship point of view.

Creating diskless devices

The `symconfigure` command supports various control operations (creation, configuration, convert, and delete) for diskless devices, using the following device type designations:

- DLDEV
- RDF1+DLDEV
- RDF2+DLDEV

- RDF21+DLDEV

Using the `symconfigure` command you can create a diskless device using the existing `create/configure dev` command with one of the previous device types.

Note: For more information about the `symconfigure` control actions, see the *Solutions Enabler Symmetrix Array Controls CLI Product Guide*.

You cannot create an RDF21+DLDEV device directly. Use the `add rdf mirror` command with `symconfigure` to create R21 diskless devices as described next.

The `set dev` command with `symconfigure` supports setting attributes on diskless devices.

Adding an SRDF mirror

The `add RDF mirror` command adds both static and dynamic SRDF mirrors to diskless devices. The procedure for setting up a diskless R21 device is no different than setting up any other type of R21 device. Before you add, the diskless (*SymDevName*) device must already be an RDF1+DLDEV or an RDF2+DLDEV device as shown in Figure 25:

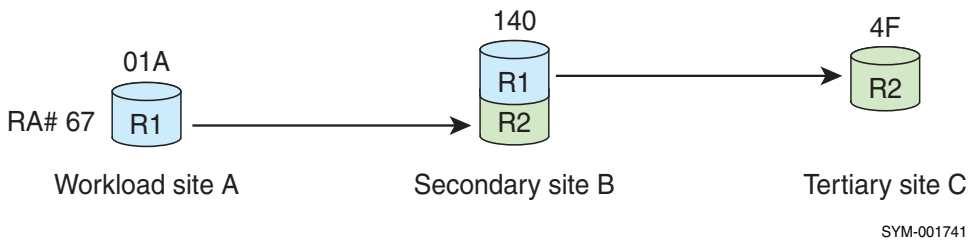


Figure 25 Adding a diskless SRDF mirror

In the command file syntax for `symconfigure`, add the R21 mirrors in two steps as follows:

From the site A example:

```
add rdf mirror to dev 01A
  ra_group=67, mirror_type=RDF1
  remote_dev=140
...
```

From the site C example:

```
add rdf mirror to dev 04F
  ra_group=67, mirror_type=RDF2
  remote_dev=140
```

Restrictions

The following are restrictions for adding an SRDF mirror:

- Either the local or the remote device can be diskless, however, both the local and the remote SRDF device cannot be diskless.
- Diskless devices can only be configured on a fibre or GigE SRDF directors.
- Cannot add a mix of diskless and non-diskless SRDF devices to an SRDF group with devices in Async mode.

- The create pair action is blocked if it results in an R1->R21->R2 relationship where the R1 and the R2 are diskless devices.
- When configuring a diskless device the modes should be set as per rules discussed in "SRDF control and set operations" on page 165.

Viewing diskless devices

The `symdev list` and `symrdf list` commands provide specific options to facilitate viewing diskless devices in your environment.

symrdf list

The `symrdf list` command supports a `-diskless_rdf` option to view SRDF diskless devices. You can use the `-diskless_rdf` option with `-R1`, `-R2`, `-R21`, or `-dynamic`, and the output returns the requested diskless SRDF or SRDF capable devices.

For example:

```
symrdf list -diskless_rdf
```

symdev list

When the `-dldev` option is included in the `symdev list` command, all configured diskless devices are in the output, as shown in the following display. You can specify the `-dldev` option with `-R1, -R2, -R21`, or `-dynamic`, and the output returns the requested diskless SRDF or SRDF capable devices.

For example:

```
symdev list -sid 305 -dldev
```

```
Symmetrix ID: 000192600305

        Device Name          Directors                   Device
--------------------------- ------------- -------------------------------------
                                                                          Cap
Attribute    Sts   (MB)
--------------------------- ------------- -------------------------------------

07A0 Not Visible            ???:? ???:?   RDF21+DLDEV    Grp'd      RW    1031
07A1 Not Visible            ???:? ???:?   RDF21+DLDEV    Grp'd      RW    1031
```

symdev show

The `symdev show` command displays information related to diskless devices:

- The `Device Configuration` field shows the device as being an R21 diskless device.
- The `Device SA Status` will always show as N/A because diskless devices can not be mapped to a host.
- Each `RDF Information` section displays a `Paired with Diskless Device` field to indicate if the device is in an SRDF relationship with a diskless SRDF device. The `Paired with Diskless Device` field reflects the device type for the SRDF partner of this device.

```
symdev show 07A0 -sid 05(for a R21 diskless device)
.
```

```
    Device Configuration     : RDF21+DLDEV      (Non-Exclusive Access)
.
.
Device Status            : Ready            (RW)

    Device SA Status         : N/A              (N/A)

    Mirror Set Type          : [R2 Remote,R1 Remote,N/A,N/A]

    Mirror Set DA Status     : [RW,RW,N/A,N/A]

    Mirror Set Inv. Tracks   : [0,0,0,0]

    Back End Disk Director Information
        {
        Hyper Type                              : R2 Remote
        Hyper Status                            : Ready           (RW)
        Disk [Director, Interface, TID]         : [N/A,N/A,N/A]
        Disk Director Volume Number             : N/A
        Hyper Number                            : N/A
        Mirror Number                           : 1

        Hyper Type                              : R1 Remote
        Hyper Status                            : Ready           (RW)
        Disk [Director, Interface, TID]         : [N/A,N/A,N/A]
        Disk Director Volume Number             : N/A
        Hyper Number                            : N/A
        Mirror Number                           : 2
...
}
    RDF Information
        {
        Device Symmetrix Name                   : 7A0
        RDF Type                                : R2
        RDF (RA) Group Num                      : 210             (D1)

        Remote Device Symmetrix Name            : 0380
        Remote Symmetrix ID                     : 000192600284

        R2 Device Is Larger Than The R1 Device  : False
        Paired with Diskless Device             : False
        Concurrent RDF Relationship             : False
        Cascaded RDF Relationship               : True
...
    RDF Information
        {
        Device Symmetrix Name                   : 7A0
        RDF Type                                : R1
        RDF (RA) Group Num                      : 230             (E5)

        Remote Device Symmetrix Name            : 3A0
        Remote Symmetrix ID                     : 000192600282

        R2 Device Is Larger Than The R1 Device  : False
        Paired with Diskless Device             : False
        Paired with a Concurrent RDF Device     : False
        Paired with a Cascaded RDF Device       : False
...
```

IMPORTANT

The `symcg`, `symdg`, or `symdev` commands fail if used with `relabel` when the scope includes any diskless devices.

For more information about `symdev list` and `show` actions, see the *Solutions Enabler Symmetrix Array Management CLI Product Guide*.

Restarting a diskless SRDF environment

The following are recovery behaviors with diskless environments:

- When restarting, the R21->R2 hop will be recovered before recovering the R1->R21 hop. The dependency here is that the R1->R21 relationship cannot be RW on the link when the R21->R2 relationship is NR on the link.
- When recovering a diskless SRDF configuration, the R21->R2 relationship, the `restart_sync_type` will always be adaptive copy write pending (ADCOPY_WP) if the R21 is a diskless device.

CHAPTER 7
TimeFinder and SRDF

This chapter explains how to use TimeFinder functionality with SRDF using the SYMCLI. Topics include:

- TimeFinder consistent splits across SRDF 172
- Multi-hop operations 173

TimeFinder consistent splits across SRDF

TimeFinder consistent split allows you to split off a consistent, restartable copy of a database management system within seconds with no interruption to online service. A concurrent split helps to avoid inconsistencies and restart problems that can occur when splitting a database-related BCV without first quiescing the database. Consistent split operations are implemented using the Enginuity Consistency Assist feature.

Consistent split operations can also be used in conjunction with SRDF Automated Replication (SRDF/AR) to set up automatic remote mirroring according to a predefined copy schedule. SRDF/AR is explained in "SRDF/Automated Replication" on page 179.

Enginuity Consistency Assist

TimeFinder consistent split operations are accomplished using the `-consistent` option with the `symmir` command. For more information about the `symmir` command, refer to the *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*.

The `-consistent` option can also used with the `symreplicate` command to run a copy cycle, which freezes I/O to all devices in a device or composite group for both single-hop and multi-hop configurations.

To consistently split BCV pairs using Enginuity Consistency Assist (ECA), you must have either a control host with no database or a database host with a dedicated channel.

Refer to Figure 26 on page 173 for a depiction of how a control host can perform ECA consistent splits for three database hosts that access devices on a Symmetrix array.

Device or composite groups must be created on the controlling host for the target database to be consistently split. These groups can be created to include all of the devices being accessed or defined by database host access.

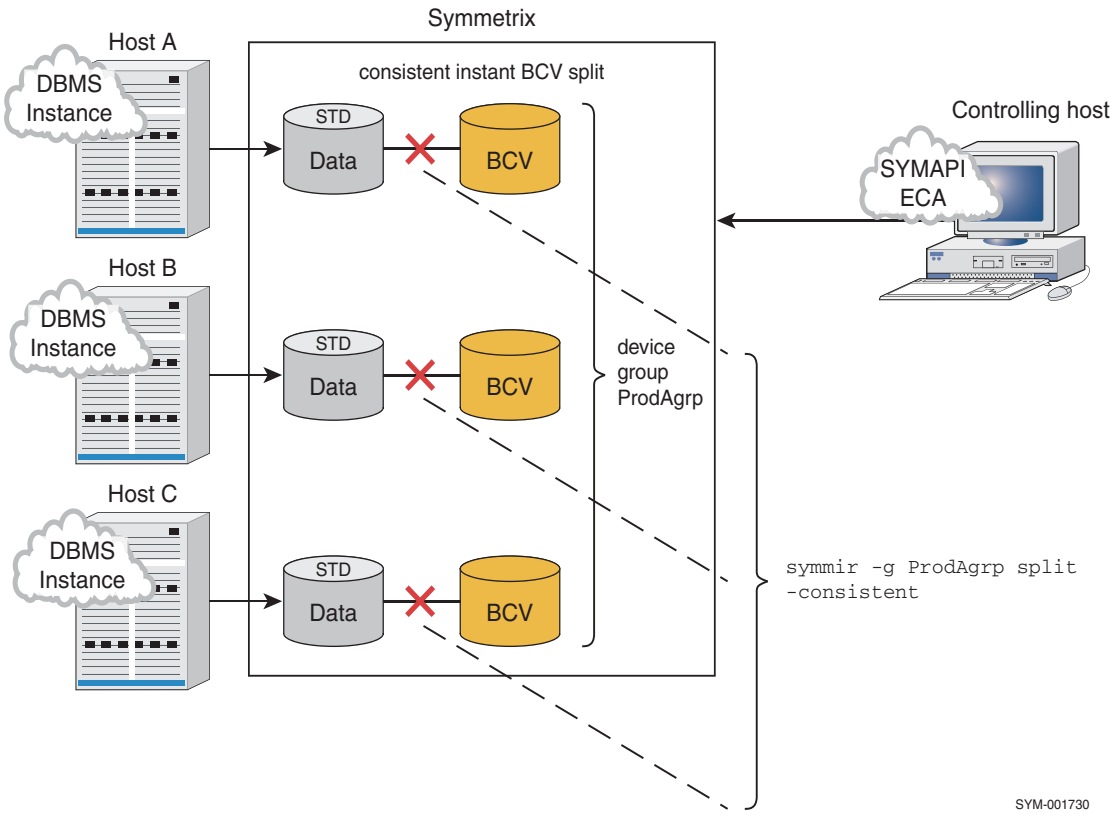


Figure 26 ECA consistent split

For example, if you define a device group that includes all of the devices being accessed by Hosts A, B, and C, then you can consistently split all of the BCV pairs related to those hosts with a single command.

Multi-hop operations

Various compounded remote configurations can be managed by your host using both the TimeFinder and SRDF components of SYMCLI.

As shown in Figure 27 on page 178, you can have multiple sites (for example, remote Sites B and C) on SRDF links to remotely mirror a local Symmetrix array at Site A. Remote Site B, functioning as a remote mirror to the standard devices at Site A, is most typical. You then can have a third site on an SRDF link (remote Site C) to remotely mirror just the BCV devices in the Symmetrix array at Site A.

Multi-hop SRDF sites

You can also multi-hop to a second-level SRDF where Remote Site D functions as a remote mirror to the standard devices of Site A and Remote Site E remotely mirroring Site A's BCV.

Command `symrdf` manages the SRDF pairs within the SRDF link while `symmir` manages the BCV pairs within any one site.

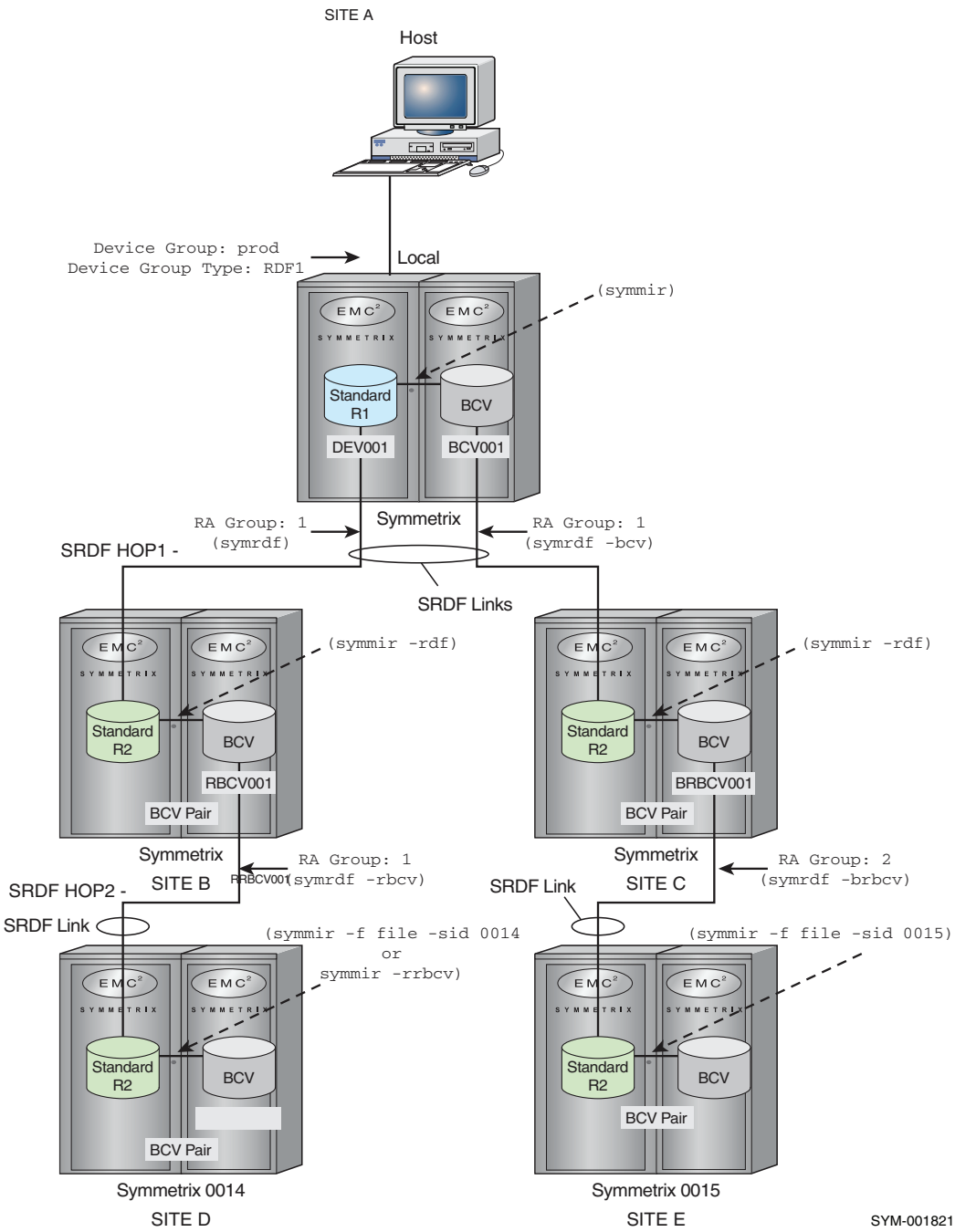
System-wide device groups

Before you begin applying any `symmir` operations, you must be working with an existing group of SRDF devices. To create a device group containing STD and BCV RDF1 devices, type:

```
symdg create prod -type RDF1
symdg -g prod add dev 0001 -sid 344402 DEV001
symbcv -g prod associate dev 000A BCV001
symbcv -g prod associate dev 000C -rdf RBCV001
symbcv -g prod associate dev 0009 -bcv -rdf BRBCV001
symbcv -g prod associate dev 0004 -rrdf RRBCV001
```

At this point, all these devices must be established with the `symmir` and `symrdf` commands.

The following set of examples illustrate how various `symmir` and `symrdf` commands might be applied to split operations throughout a complex remote configuration such as one shown in Figure 27 on page 178.



Examples of system-wide splits

Before you begin applying any `symmir` and `symrdf` operations, you must be working with an established group of SRDF devices. To split the BCV pair within Site A, type:

```
symmir -g prod split
```

To split SRDF pairs at Site B from host-connected Site A, type:

```
symrdf -g prod split
```

To split the BCV pairs within Site B, type:

```
symmir -g prod -rdf split
```

To split BCV SRDF pairs at Site C from host-connected Site A, type:

```
symrdf -g prod -bcv split
```

To split the BCV pairs within Site C, type:

```
symmir -g prod -rdf -bcv split
```

To split BCV SRDF pairs at Site D from host standard-associated Site B, type:

```
symrdf -g prod -rbcv split
```

To split the BCV pairs within Site D, type:

```
symmir -f dfile -sid 0014 split
```

or

```
symmir -g prod -rrbcv split
```

To split BCV SRDF pairs at Site E from host BCV-associated Site C, type:

```
symrdf -g prod -brbcv split
```

To split the BCV pairs within Site E (hop 2), type:

```
symmir -f dfile -sid 0015 split
```

Other operations, such as `establish` and `restore`, apply and execute in the same manner for the remote sites.

For more information about `symmir`, refer to the *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*.

Targeting commands to various multi-hop devices and links

This section describes the commands to target the various devices and links in complex multi-hop SRDF environments.

Table 14 shows a sequence of command steps for some basic control operations, which touch every device and SRDF link in a complex multi-hop configuration. This table works with and is illustrated by Figure 27 on page 178. The following numbering of commands directly associates with the bubble numbers shown in Figure 27.

Table 14 Remote multi-hop SRDF commands (page 1 of 2)

Step	CLI control operation	Description
1	symrdf -g ‹› establish	Creates the standard associated hop 1 copy.
2	symmir -g ‹› split -rdf	Splits the standard associated hop 1 BCV device pair.
3	symrdf -g ‹› establish -rbcv	Creates the standard associated hop 2 copy.
4	symrdf -g ‹› restore -rbcv	Restores the standard associated hop 1 BCV with the hop 2 copy.

Table 14 Remote multi-hop SRDF commands (page 2 of 2)

Step	CLI control operation	Description
5	symmir -g ‹› restore -rdf	Restores the standard associated hop 1 copy with the hop 1 BCV.
6	symrdf -g ‹› restore	Restores the standard device with the hop 1 copy.
7	symmir -g ‹› split	Splits the standard/BCV pair.
8	symrdf -g ‹› establish -bcv	Creates the BCV associated hop 1 remote copy.
9	symmir -g ‹› split -rdf -bcv	Splits the BCV associated hop 1 device pair.
10	symrdf -g ‹› establish -brbcv	Creates the BCV associated hop 2 copy.
11	symrdf -g ‹› restore -brbcv	Restores the BCV associated hop 1 BCV with the hop 2 copy.
12	symmir -g ‹› restore -rdf -bcv	Restores the standard device associated hop 1 copy with the hop 1 BCV.
13	symrdf -g ‹› restore -bcv	Restores the BCV device with the hop 1 copy.
14	symmir -g ‹› restore	Restores the standard device with the BCV copy.
15	symmir -f ‹› -sid 056 establish or symmir -g ‹› -rrbcv establish	Creates the BCV associated hop 2 BCV copy.
16	symmir -f ‹› -sid 056 split or symmir -g ‹› -rrbcv	Splits the BCV-associated hop 2 device pair.

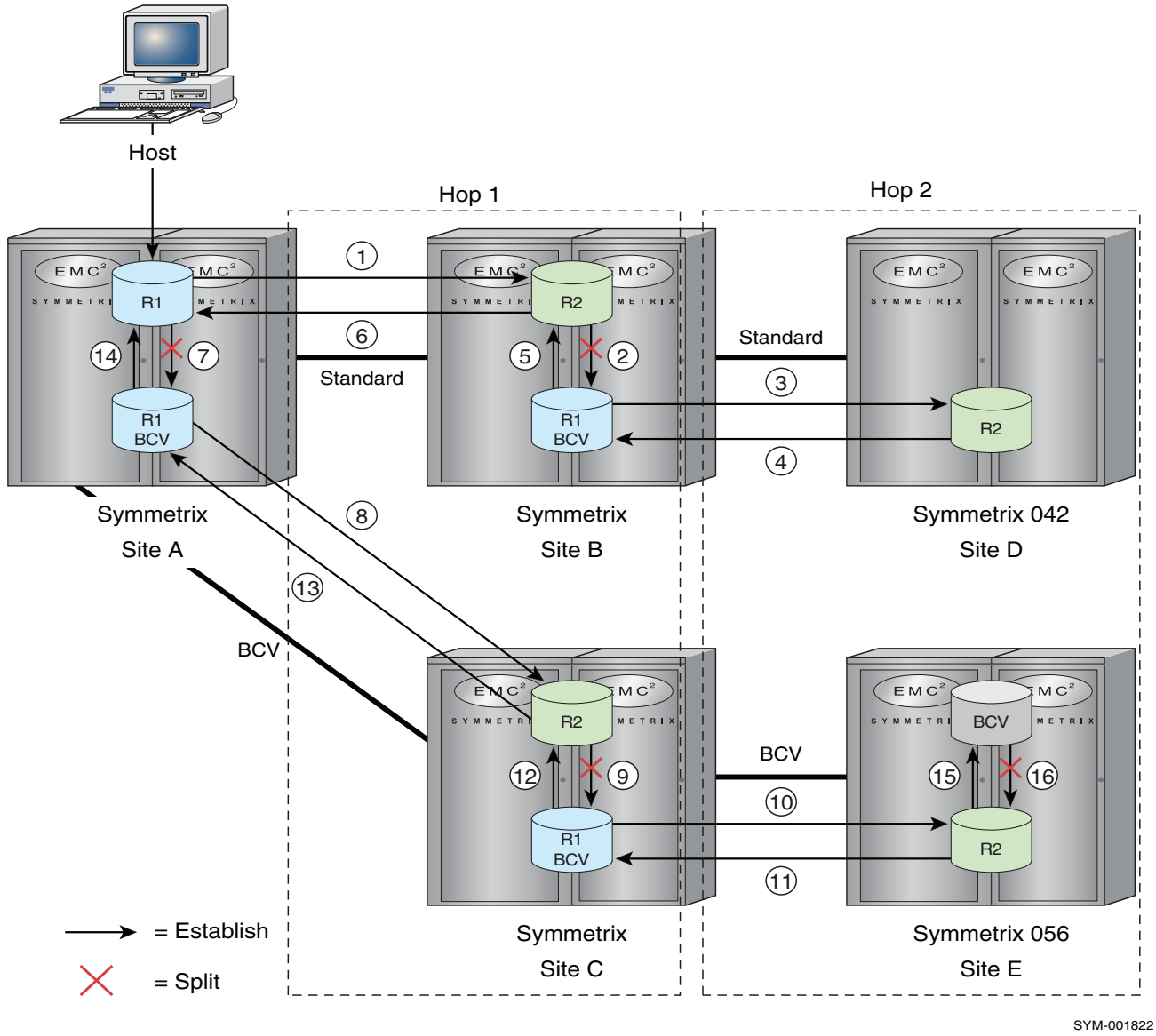


Figure 27 Remote multi-hop SRDF configurations

CHAPTER 8
SRDF/Automated Replication

This chapter explains how to use the SYMCLI `symreplicate` command to invoke the SRDF Automated Replication (SRDF/AR) functionality. Topics include:

- Overview 180
- SRDF/Automated Replication operations 181

Overview

The `symreplicate` command invokes the SRDF Automated Replication (SRDF/AR) facility. The command performs automated consistent replication of data from standard devices and from RDF1 BCV devices over SRDF links to the remote SRDF pair.

- For Symmetrix DMX systems and Enginuity versions 5773 and lower, SRDF/AR uses either native TimeFinder/Mirror or TimeFinder Clone Emulation mode.
- For Symmetrix VMAX family arrays and Enginuity versions 5874 and higher, SRDF/AR uses TimeFinder Clone Emulation mode.

By default, the replication process is performed in the background. The `symreplicate` command supports both single-hop and multi-hop SRDF configurations. You can start, stop, or restart a `symreplicate` session without degradation of the data copy. During a `symreplicate` session, you can have access to an independent copy of the replicating data by setting up a concurrent BCV.

Restrictions

Review the following restrictions before using SRDF/AR:

- Does not support SRDF/Asynchronous-capable devices
- The `symreplicate` command can only operate against an entire device group or composite group, and cannot limit its scope to a specific SRDF group using the `-rdfg` option
- When running `symreplicate` against device groups and composite groups of the ANY type, the following restrictions apply:
  - Concurrent and cascaded SRDF devices are not supported for device groups (DG) or composite groups (CG).
  - The following combinations of standard devices are supported when using the `-consistent` option:
    - All STDs are non-SRDF
    - All STDs are R1 devices
    - All STDs are R2 devices
    - STDs contain a mixture of R1s and non-SRDF devices
    - STDs contain a mixture of R2 and non-SRDF devices

Note: Device external locks in the Symmetrix array are held during the entire `symreplicate` session, which is necessary to block other applications from altering device states while the session executes. For more information, refer to "Locked devices" on page 197.

SRDF/Automated Replication operations

This section describes the SRDF/AR control operations.

Single-hop data copies

Figure 28 shows how `symreplicate` copies data in a single-hop configuration for a complete copy cycle:

- From the standard device to the BCV of the local Symmetrix array (1a path).
- From the BCV device of the local Symmetrix array to the standard device of the remote Symmetrix array (1b path).
- From the remote standard device to its BRBCV device (1c path).

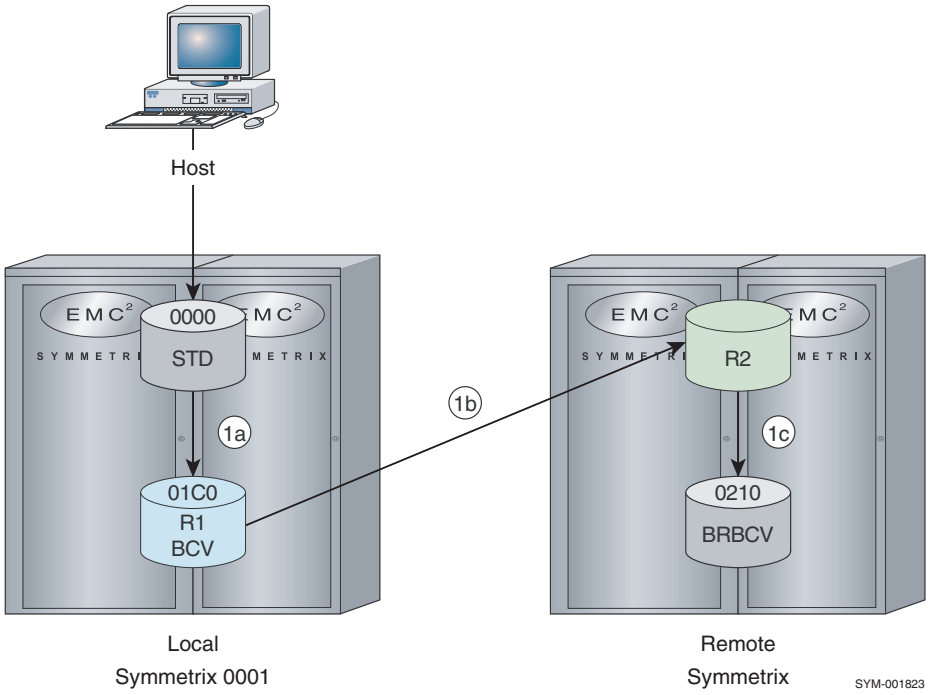


Figure 28 Automated data copy path in single-hop SRDF systems

To choose a single-hop `symreplicate` session, you must set the replication type parameter in the replicate options file (refer to "Setting the symreplicate file parameters" on page 192) as follows:

```
SYMCLI_REPLICATE_HOP_TYPE=SINGLE
```

The `symreplicate` session incrementally establishes SRDF and BCV pairs, and then differentially splits BCV pairs to reduce data transfers.

For expanded operational examples of SRDF/AR in a single-hop configuration, you can refer to Chapter 16, "Performing SRDF/Automated Replication Operations."

Setup for a single-hop data replication

The single-hop data replication copies data from the local Symmetrix array to the remote Symmetrix array.

To set up a single-hop data `symreplicate` session:

- Select any number of standard devices of the same type (R1, R2, or non-SRDF).
- Create a device group or composite group of the same type.
- Add the devices, and associate an equal number of R1-BCV devices of matching sizes.
- Associate an equal number of BRBCV devices (remote BCVs), also of matching sizes.

The following command sequence illustrates this setup:

```
symdg create newdg
symdg add dev 0000 -g newdg -sid 35002
symdg add dev 0001 -g newdg
    .
    .
    .
symbcv associate dev 01C0 -g newdg
symbcv associate dev 01C1 -g newdg
    .
    .
    .
symbcv associate dev 0210 -g newdg -bcv -rdf
symbcv associate dev 0211 -g newdg -bcv -rdf
    .
    .
    .
```

The `symreplicate` command supports the use of composite groups (`-cg`) to implement single-hop or multi-hop configurations for devices that span multiple Symmetrix arrays.

Before starting a `symreplicate` session, the following conditions must be met:

- Both sets of BCV pairs must have a pairing relationship.
- The local BCV pairs must be established, the SRDF pairs must be in the Suspended pair state, and the remote BCVs (BRBCVs) must be in the split pair state. Ensure there are no writes allowed to the BRBCV by any directly attached host at the remote site.

Pair state auto setup for SRDF/AR

The pair state setup for SRDF/AR can be achieved automatically by using either the `symreplicate setup` command or the `-setup` option with the `symreplicate start` command.

The auto-replication setup action sets up the required pair states for devices and executes one copy (auto-replication) cycle. By setting up the device states ahead of time, replication processing time is saved. The symreplicate options file defines the hop type (single or multi) and any copy cycle parameters to use for the `setup` and `start` commands. The `setup` command executes only one cycle of the `symreplicate` session, regardless of the number of cycles defined in the options file, and then exits.

Note: If you prefer, these conditions can be established by manually reproducing the single-hop replication cycle through a sequence of SRDF and TimeFinder CLI commands. For more information on how to manually set up the single-hop replication environment, refer to "Manual setup for a single hop" on page 184. Or, for more information on how to manually set up the multi-hop replication, refer to "Manual setup for multi-hop" on page 186.

Note that the setup operation only corrects pair states of devices in the group. If a BCV in the group is paired with a standard device outside of the group, setup does not correct it.

The following command shows how to execute the `symreplicate setup` command on a device group (`DevGrp1`) using an options file (`OpFile`):

```
symreplicate -g DevGrp1 setup -options Opfile
```

Note: The setup command may take some time to run to completion and does not exit until the devices are in the required pair state to run the `symreplicate` session.

For more information on the available parameters that can be defined in the replicate options file, refer to "Setting the symreplicate file parameters" on page 192.

When executing the `symreplicate start` command with the `-setup` option, the first cycle puts the devices in the required pair state. The following command line shows how to execute the `symreplicate start` command with the `-setup` option:

```
symreplicate -g DevGrp1 start -options Opfile -setup
```

The default setup operation (using either the `setup` action or the `-setup` option) provides no I/O optimization, and does not engage any special algorithm changes in the selection of pair assignments. For standard devices encountered without BCVs, the first unassigned BCV device found is paired with the standard.

Exact initial pairing

Use the `-exact` option to start the `symreplicate` session with the STD-BCV pair relationships in the exact order that they were associated/added to the device group or composite group.

Optimizing I/O with pair assignment

Use the `-optimize` option (when you use the `-setup` option or the `setup` argument) to optimize the disk I/O on standard/BCV pairs in the device or composite group.

This will cause the setup action to split all pairs and perform an optimized STD-BCV pairing within the specified group. For device groups using the optimize option, the device pair selection attempts to pair devices in the group that are not on the same disk adapter to distribute the I/O. For example:

```
symreplicate setup -g DgName -optimize
```

For composite groups, the same optimize pairing behavior can be targeted to a Symmetrix RA group. For pair assignment in RA groups that provides remote I/O optimization (distribution by using different remote disk adaptors), use the `-optimize_rag` option with either the `-setup` option or the `setup` argument. For example:

```
symreplicate setup -cg CgName -optimize_rag
```

Note: Single-hop replication does a full optimization on all RA groups.

Consistent split option

Using the `-consistent` option with the `start` action, you can consistently split all of the BCV pairs on the local Symmetrix array for a typical SRDF configuration, or on the Hop 1 remote Symmetrix array for a multi-hop configuration. This also requires a TimeFinder/CG license.

Consistent split operations are automatically retried if the split fails to complete within the allotted timing window. If a consistent split operation fails due to the consistency timing window closing before the split can complete (`SYMAPI_C_CONSISTENCY_WINDOW_CLOSED`), then the first-hop local BCV device pairs are automatically resynchronized and the split operation will be reattempted. The consistent split error recovery operation will be attempted the number of times specified in the `SYMCLI_REPLICATE_CONS_SPLIT_RETRY` file parameter, which is defined in the replicate options file. If a value is not specified, then the recovery operation will be attempted 3 times before terminating the `symreplicate` session.

For more information on the available parameters that can be defined in the replicate options file, refer to "Setting the symreplicate file parameters" on page 192.

Manual setup for a single hop

If you prefer, these conditions can be established by manually reproducing the single-hop replication cycle through a sequence of SRDF and TimeFinder CLI commands. The following are the manual single-hop replication steps:

1. After waiting for any ongoing establish to complete, split the BCV pairs:

   `symmir split -g newdg`

2. Establish the SRDF pairs:

   `symrdf establish -g newdg -bcv`

3. After waiting for any ongoing establish to complete, suspend the SRDF pairs:

   `symrdf suspend -g newdg -bcv`

4. Establish the BCV pairs:

   `symmir establish -g newdg -exact`

5. Establish the remote BRBCV pairs:

   `symmir establish -g newdg -bcv -rdf -exact`

6. After waiting for any ongoing establish to complete, split the remote BRBCV pairs:

   `symmir split -g newdg -bcv -rdf`

Note: You may have to include additional command options in some of the above steps (for example, `establish -full` for BCV pairs without relationships).

Multi-hop data copies

As shown in Figure 29, for a multi-hop configuration in a complete copy cycle, `symreplicate` copies data:

- From the local standard device to the standard of the remote Hop 1 Symmetrix (2a path)
- From the Hop 1 standard device to its BCV (RBCV) (2b path)
- From the Hop 1 RBCV device to the standard device of Hop 2 Symmetrix (2c path)
- From the Hop 2 standard device to its BCV (RRBCV) (2d path)[1]

To choose a multi-hop `symreplicate` session, you must set the replication type parameter in the replicate options file (refer to "Setting the symreplicate file parameters" on page 192) as follows:

```
SYMCLI_REPLICATE_HOP_TYPE=MULTI
```

Note: If you do not want the final Hop 2 BCV updated, you can set `SYMCLI_REPLICATE_USE_FINAL_BCV=FALSE` in the replicate options file.

For expanded operational examples of SRDF/AR in a multi-hop configuration, see Chapter 16, "Performing SRDF/Automated Replication Operations."

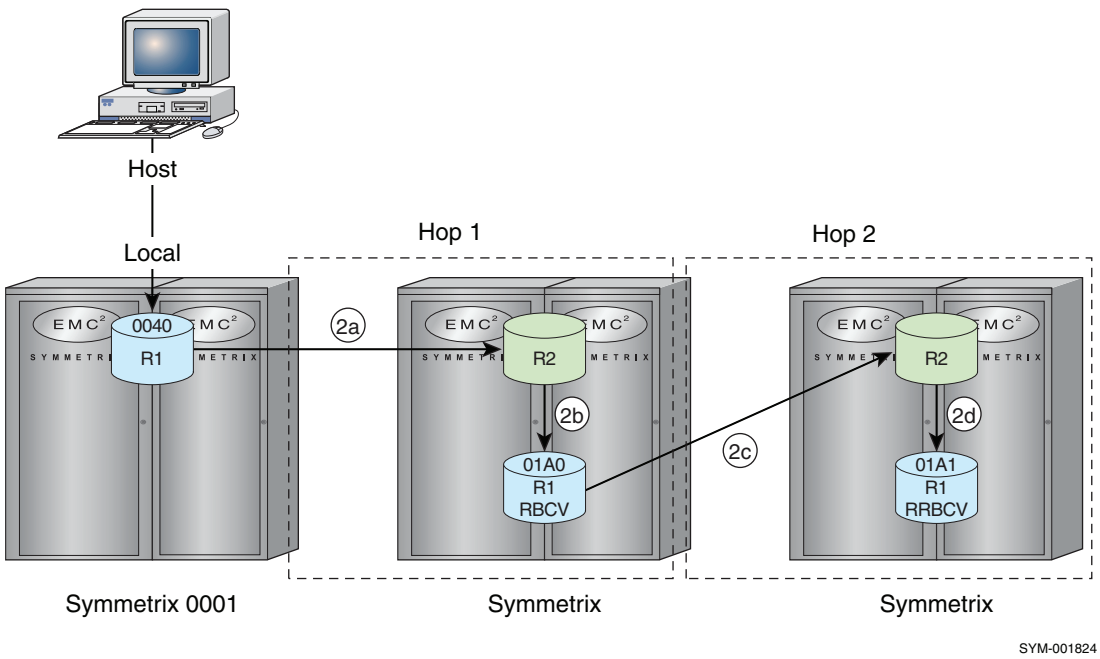


Figure 29 Automated data copy path in multi-hop SRDF systems

1. Applies only when you have a final BCV in this Hop 2 Symmetrix path and you have not disabled it.

Setup for a multi-hop data replication

Multi-hop data replication copies data from the local Symmetrix array to the Hop 1 remote Symmetrix array, and then to the Hop 2 Symmetrix array. To set up a multi-hop data `symreplicate` session:

- Create an R1 device group (`-g`) or composite group (`-cg`),
- Add any number of R1 devices.
- Remotely associate an equal number of matching sized R1-BCVs or Hop 1 RBCV devices.

The following command sequence illustrates this setup:

```
symdg create newdg2 -type RDF1
symdg add dev 0040 -g newdg2 -sid 0001
    .
    .
symbcv associate dev 01A0 -g newdg2 -rdf
symbcv associate dev 01A1 -g newdg2 -rrdf
```

Before starting a `symreplicate` session without a setup operation, the local SRDF pairs must be synchronized, the BCV pairs must be established, and the remote SRDF pairs must be suspended. If the final BCVs in the second-hop Symmetrix array are used, the BCVs must be in the split state.

Optionally, the device pair state can be configured automatically by using the `symreplicate setup` command or the `-setup` option with the `symreplicate start` command. Refer to "Pair state auto setup for SRDF/AR" on page 182 for information on automating setup conditions.

Manual setup for multi-hop

Manual setups for these conditions can be established by manually reproducing the multi-hop replication cycle through a sequence of TimeFinder `symmir` CLI commands. The following are the manual multi-hop replication steps for the configuration in Figure 29 on page 185:

1. After waiting for any ongoing establish to complete, split the BCV pairs (point 2b):

   ```
   symmir split -g newdg2 -rdf -remote
   ```

   Establish the remote SRDF pairs (first hop BCV with R2 second hop at point 2c). (This step was accomplished in this last command by the use of the `-remote` option.)

2. After waiting for the SRDF establish to complete, suspend the remote SRDF pairs (2c), and establish the BCV pairs (2b):

   ```
   symmir establish -g newdg2 -rdf -exact
   ```

3. Establish the BCV pairs in the second Symmetrix hop (2d) by using either a device file or the `-rrbcv` command option:

   ```
   symmir establish -f 2nd_hop_devs.txt -sid SymmID
   ```

   or

   ```
   symmir establish -g newdg2 -rrbc
   ```

Note: To use the `-rrbcv` option, the SRDF BCV devices must have been previously associated with the group, using `symbcv -rrdf`.

4. After waiting for any ongoing establish to complete, split the 2nd hop BCV pairs:

   ```
   symmir split -f 2nd_hop_devs.txt
   ```

   or

   ```
   symmir split -g newdg2 -rrbcv
   ```

   Note: Steps 3 and 4 are performed when you want the final hop 2 BCVs to be used in the replicate cycle.

Note: You may have to include additional command options in some of the above steps (such as `establish -full` for BCV pairs without relationships).

The `-preaction` and `-postaction` options can be used to specify scripts for `symreplicate` to run before and after step 1 (splitting the BCVs).

Concurrent BCVs with SRDF/AR

If you require an independent copy of your data during a replication cycle, you can set up concurrent BCVs. One BCV copy is associated with the SRDF/AR device group and the other BCV copy is not. The BCV not associated with the replication cycle receives the same data as the one associated with the SRDF/AR devices. The non-SRDF/AR BCV can be accessed by its host during the `symreplicate` cycle.

Refer to Figure 30 on page 188 for a depiction of how to set up concurrent BCVs in a multi-hop configuration.

Devices 0027 and 0039 are not part of the SRDF/AR copy cycle. To access these devices from the production host during the SRDF/AR copy cycle, you must define separate device files on the host that include the standard R2 device and the R2 BCV on Hop 1 and Hop 2. The device files are then used to establish the BCV pairs, split them, and access the BCV devices.

For an illustrated example, refer to Chapter 16, “Performing SRDF/Automated Replication Operations.”

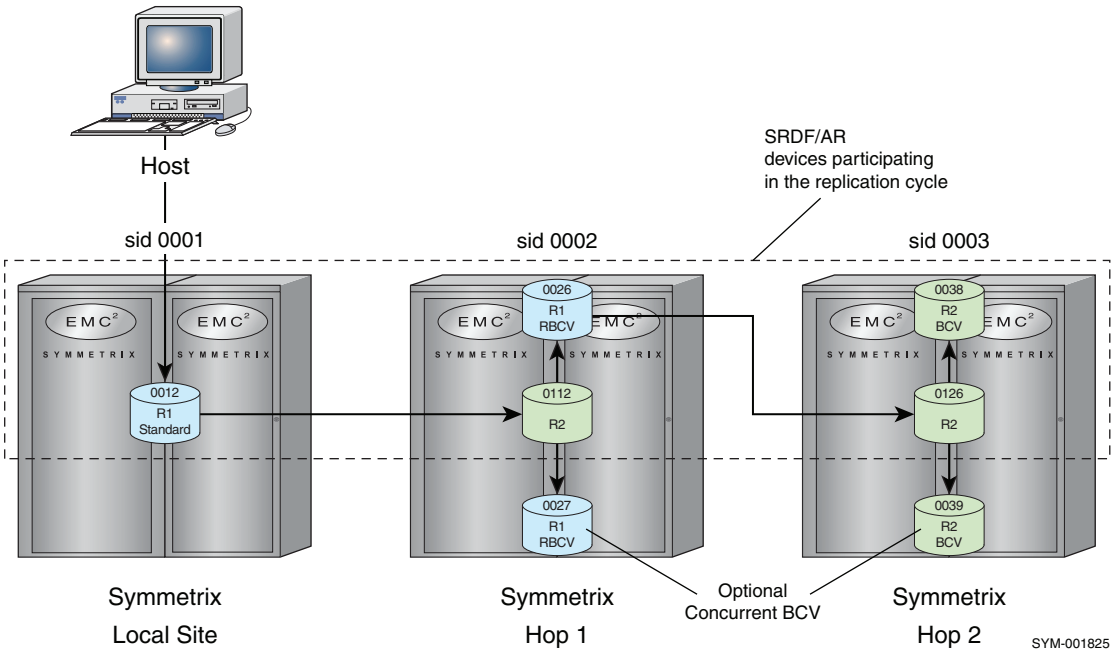


Figure 30 Concurrent BCV in a multi-hop configuration

Note: For expanded SRDF/AR operational examples using concurrent BCVs in both the single-hop and multi-hop configurations, you can refer to Chapter 16, "Performing SRDF/Automated Replication Operations."

Replication cycle patterns

You can manipulate the replication cycle patterns to fit your site's needs by setting the following parameters in the symreplicate options file (refer to "Setting the symreplicate file parameters" on page 192 for options file syntax) as follows:

SYMCLI_REPLICATE_CYCLE=*CycleTime*

CycleTime is a timer that indicates the period of time in *minutes* or *hours*:*minutes*(*hh*:*mm*) between when each copy action starts and when it starts again (how often the copy reoccurs). For example, a *CycleTime* of 120 would initiate a new copy every 2 hours.

SYMCLI_REPLICATE_NUM_CYCLES=*NumCycles*

NumCycles indicates the number of replication cycles (copies) to perform before `symreplicate` exits. For example, a zero value (the default value) results in continuous cycling until the `symreplicate stop` command is issued.

SYMCLI_REPLICATE_CYCLE_DELAY=*Delay*

Delay indicates the minimum amount of time to wait between the end of one copy cycle and the beginning of the next. For example, a *Delay* of 20 would always force a wait of 20 minutes or more between cycles.

SYMCLI_REPLICATE_CYCLE_OVERFLOW=*OvfMethod*

OvfMethod indicates how to behave when the actual copy time of your data and/or data transfer throughput is so large as to exceed the *CycleTime* value. Here, the initial copy event has overflowed into the period that should be for the next copy cycle. Possible behavior values are:

`IMMEDIATE` — When overflowed, starts a new cycle immediately after the current copy finishes.

`NEXT` — When overflowed, waits for the copy to finish, and then starts at the next expiration time (*CycleTime*). (Starts the copies on multiples of the *CycleTime* parameter.)

For example, if a 1-hour copy cycle completed in 1.5 hours, the next cycle could be set to begin immediately (`IMMEDIATE`) or in half an hour (`NEXT`).

First time cycle parameters

Choosing all the exact cycle time parameters (described in the previous section) may not be possible the first time. A basic replication parameter strategy is to loosely select time constraints, and then tighten the parameters at some point later when you have a sense of data size and SRDF throughput expectations.

Table 15 provides two possible parameter setups for an initial `symreplicate` session trial:

Table 15 Initial setups for cycle timing parameters

SYMCLI_REPLICATE_CYCLE=60 SYMCLI_REPLICATE_CYCLE_DELAY=0 SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT	Every hour if possible, or every 2, or 3 hours based on data throughput and size.
SYMCLI_REPLICATE_CYCLE=0 SYMCLI_REPLICATE_CYCLE_DELAY=60	Cycle through the first copy, then wait 60 minutes (delay), and then another cycle, delay, and so on.

Start the `symreplicate` session with the basic parameters set and run `symreplicate query` to monitor session progress, noting the timing results of the initial copies. Then, adjust the various timing parameters to best accommodate the copy requirements for your needs.

Cycle time and invalid track statistics

You can display statistical information for cycle time and invalid tracks by using the `symreplicate stats` command. The command can be issued by device group (`-g`) or composite group (`-cg`) for a specified Symmetrix (`-sid`) and information can optionally be written to a specified log file (`-log`). Cycle time (`-cycle`) statistics will be displayed for the last SRDF/AR cycle time, the maximum cycle time and the average cycle time. Invalid track (`-itrks`) statistics will be displayed for the last SRDF/AR cycle, the maximum invalid tracks and the average number of invalid tracks per SRDF/AR cycle. The `-all` option is the default and will display both the cycle time and invalid tracks statistics.

For example, to display both cycle time and invalid track statistics for device group `srdfar` on Symmetrix 1123, type:

```
symreplicate -g srdfar -sid 123 -all stats
```

```
Group Name: srdfar

Cycle Time (hh.mm.ss):
---------------------------------------

Last Cycle Time:  06:10:01
Max Cycle Time:   08:00:00
Avg Cycle time:   06:00:00

Invalid Tracks:
---------------------------------------

Last Cycle:  12345 ( 9055.5 MB)
Maximum:     10780 ( 8502.3 MB)
Average:     11562 ( 7500.0 MB)
```

Replication log entries

You can track the steps in a `symreplicate` session by setting the `SYMCLI_REPLICATE_LOG_STEP` entry in the options file to `TRUE`. This option causes `symreplicate` to write an entry to the SYMAPI log file after each step is completed. Log entries contain the time that the step ended and whether it was successful.

Refer to "Setting the symreplicate file parameters" on page 192 for the options file syntax.

Clustered SRDF/AR environments

Symmetrix arrays support clustered SRDF/AR environments for multiple node (host) capability. Clustered SRDF/AR provides the capability to start, stop, and restart `symreplicate` sessions from any host connected to any local Symmetrix array participating in the `symreplicate` session.

The clustered SRDF/AR environment allows the replication log file to be written directly to the Symmetrix File System (SFS) instead of the local host directory of the node that began the session. If the primary node should fail, then any locally attached host to the Symmetrix array containing the log file would then be able to restart the SRDF/AR session from where it left off.

To write the log file to the SFS, you must specify the ID of the Symmetrix array (`-sid`) where the log file is to be stored at the start of the `symreplicate` session, along with a group name (`-g`, `-cg`) and an optional user log filename (`-log`). For example:

```
symreplicate start -g session1 -log srdfar1.log -sid 201
```

Note: Not specifying the Symmetrix ID (`-sid`) at the start of the session, causes the log file to be written to local disk using the default SYMAPI log directory, which is not restartable from another node.

If you begin a session and specify a user log file name (`-log`), you must specify the `-log` option for all other commands in the session sequence.

If you begin a session specifying only the group name (`-g`, `-cg`), the log file will be named the same as the group, and must be specified using only the `-g` or `-cg` option for all other commands in the session sequence.

If the primary node fails at some point in the replication process, the SRDF/AR session can now be restarted from another local host using the following command:

```
symreplicate restart -g session1 -log srdfar1.log -sid 201 -recover
```

The `-recover` option is used to recover the device locks from the previously started session.

Note: Before using the `-recover` option, you must ensure that no other `symreplicate` session using the same devices that is currently running.

Note: The device or composite group does not need to be defined on the restart node (host).

You can display a list of the current SRDF/AR log files that have been written to the SFS by using the `list` command with the `-sid` option as follows:

```
symreplicate list -sid 201
```

By including the `-sort` option with the `list` command, you can sort the log file list by name (default) or type.

To display the information content of a particular log file using the `show` command, you must specify the log filename (`-log`) and the Symmetrix ID (`-sid`) as follows:

```
symreplicate show -log srdfar1.log -sid 201 -all
```

The `-all` option (default) is used to display all available information contained in the log, including command-line arguments (`-args`), devices (`-devs`), and options (`-opts`). Refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference* for a description of each option.

To delete a particular log file written to the SFS, you must specify either the group name (`-g`, `-cg`) or the log filename (`-log`) (depending on if you defined a user log name when you began the session) with the `delete` command as follows:

```
symreplicate delete -log srdfar1.log
```

Setting replication retry and sleep times

You can control how long and how often `symreplicate` executes certain control operations by setting the following parameters in the `symreplicate` options file as follows:

`SYMCLI_REPLICATE_GEN_TIME_LIMIT=`*`TimeLimit`*

Controls how long errors of a general nature, such as waiting for a lock, are retried.

`SYMCLI_REPLICATE_RDF_TIME_LIMIT=`*`TimeLimit`*

Controls how long to wait for SRDF devices to enter a specific state.

`SYMCLI_REPLICATE_BCV_TIME_LIMIT=`*`TimeLimit`*

Controls how long to wait for BCV devices to enter a specific state.

`SYMCLI_REPLICATE_GEN_SLEEP_TIME=`*`SleepTime`*

Controls how long `symreplicate` should sleep before retrying a general operation.

`SYMCLI_REPLICATE_RDF_SLEEP_TIME=SleepTime`

Controls the minimum time `symreplicate` should sleep before retrying an SRDF operation.

`SYMCLI_REPLICATE_BCV_SLEEP_TIME=SleepTime`

Controls the minimum time `symreplicate` should sleep before retrying a BCV operation.

`SYMCLI_REPLICATE_MAX_BCV_SLEEP_TIME_FACTOR=Factor`

Controls the maximum time that `symreplicate` sleeps before checking the BCV device state.

`SYMCLI_REPLICATE_MAX_RDF_SLEEP_TIME_FACTOR=Factor`

Controls the maximum time that `symreplicate` sleeps before checking the SRDF device state.

Refer to "Setting the symreplicate file parameters" for expanded options file descriptions and syntax.

Note: On `restart`, if you specify an options file, the following options may not be changed: `SYMCLI_REPLICATE_USE_FINAL_BCV` or `SYMCLI_REPLICATE_HOP_TYPE`. If attempted, an error message is displayed. All other options may be specified and any new values take effect immediately.

Setting the symreplicate file parameters

The `symreplicate` file is where you can set and edit required parameter entry lines to control the replicate behavior. The following are possible parameter entries and values for the options file:

`SYMCLI_REPLICATE_HOP_TYPE=<RepType>`

Defines your configured environment in which to operate the data `symreplicate` session. This parameter is not optional and must be specified. Possible *RepType* values are:

`SINGLE` — Single-hop configuration

`MULTI` —Multi-hop configuration

`SYMCLI_REPLICATE_USE_FINAL_BCV=<TRUE|FALSE>`

Indicates whether to update the BCV in the final (last) remote Symmetrix array (for multi-hop only).

`TRUE` –(default) Replicates data copy the BCV in the final (last) remote Symmetrix array.

`FALSE` -The second hop BCV devices will be omitted.

```
SYMCLI_REPLICATE_PROTECT_BCVS=
        <NONE|BOTH|LOCAL|REMOTE|FIRST_HOP|SECOND_HOP>
```

`NONE` - (default) Establishes BCV-STD pairs without the protective establish behavior, relating to two-way mirrored BCV devices.

`LOCAL` or `REMOTE` - Causes the two mirrors of the BCV to be moved or joined to the standard device.

`BOTH` - Both the local BCV mirrors and the remote BCV mirrors get joined to their standard device.

`FIRST_HOP` or `SECOND_HOP` - Performs the protect BCV establish for first or second hop devices only in a multi-hop configuration.

`SYMCLI_REPLICATE_CYCLE=<CycleTime>`

Defines the period to wait between copy operations in total *minutes* or in an *hours*:*minutes* (*hh:mm*) format.

`SYMCLI_REPLICATE_CYCLE_DELAY=<Delay>`

Specifies the minimum time to wait between adjacent cycles. Even if a cycle overruns the specified *CycleTime* and *OvfMethod* is set to `IMMEDIATE` when *Delay* is specified, the session waits this delay time before beginning another cycle.

`SYMCLI_REPLICATE_NUM_CYCLES=<NumCycles>`

Specifies the number of cycles to perform before exiting. If you specify a value of zero, the `symreplicate` session cycles forever. The *NumCycles* default value is zero.

`SYMCLI_REPLICATE_CYCLE_OVERFLOW=<OvfMethod>`

Describes what to do if the cycle overruns the specified *CycleTime*. Possible *OvfMethod* values are:

`IMMEDIATE` — Begins next cycle immediately (the default)

`NEXT` — Skips this copy cycle and wait for the next to begin

`SYMCLI_REPLICATE_LOG_STEP=<TRUE|FALSE>`

`TRUE` - Writes a log entry to the SYMAPI log file after each step of the `symreplicate` cycle is completed. The entry displays the time that the step ended and whether the step was successful.

`SYMCLI_REPLICATE_GEN_TIME_LIMIT=<TimeLimit>`

Indicates how long errors of a general nature should be retried (for example, attempting to acquire a Symmetrix array lock). Currently, the general *TimeLimit* only applies when initiating an SRDF split or establish operation.

The default general *TimeLimit* is 00:30 if not specified.

The *TimeLimit* value enables you to control how long `symreplicate` retries certain types of operations. *TimeLimit* must be specified using one of the following formats:

hh:mm — Specifies the number of hours and minutes

sss — Specifies the number of seconds

A *TimeLimit* specified as zero (0) indicates that no time limit applies, causing the operation to be retried indefinitely.

`SYMCLI_REPLICATE_RDF_TIME_LIMIT=<TimeLimit>`

Indicates how long to wait for SRDF devices to enter a specific state. For example, after successfully issuing the command to establish an R2 BCV device with the corresponding R1 standard device, `symreplicate` waits the indicated length of time for the devices to become synchronized.

The default SRDF *TimeLimit* is 04:00 if not specified.

`SYMCLI_REPLICATE_BCV_TIME_LIMIT=<TimeLimit>`

Indicates how long to wait for BCV devices to enter a specific state. For example, after successfully issuing the command to establish a BCV device with the corresponding standard device, `symreplicate` waits the indicated length of time for the devices to become synchronized.

The default BCV *TimeLimit* is 02:00 if not specified.

`SYMCLI_REPLICATE_GEN_SLEEP_TIME=<SleepTime>`

Indicates how long `symreplicate` should sleep before retrying a general operation (for example, attempting to acquire a Symmetrix array lock). Currently, the general *SleepTime* only applies when initiating an SRDF split or establish operation. The default general *SleepTime* is 10 seconds if not specified.

The *SleepTime* value enables you to control how long `symreplicate` sleeps before retrying certain types of operations. *SleepTime* must be specified using one of the following formats:

hh:mm — Specifies the number of hours and minutes

sss — Specifies the number of seconds

A *SleepTime* must be specified as greater than zero (0).

`SYMCLI_REPLICATE_RDF_SLEEP_TIME=<SleepTime>`

Indicates the minimum length of time that `symreplicate` should sleep before retrying an SRDF device operation. For example, after issuing the command to establish an R2 BCV device with the corresponding R1 standard device, `symreplicate` sleeps the indicated length of time before retrying the operation.

The default SRDF *SleepTime* is 15 seconds if not specified.

`SYMCLI_REPLICATE_BCV_SLEEP_TIME=<SleepTime>`

Indicates the minimum length of time that symreplicate should sleep before retrying a BCV device operation. For example, after issuing the command to establish a BCV device with the corresponding standard device, `symreplicate` sleeps the indicated length of time before retrying the operation.

The default BCV *SleepTime* is 10 seconds if not specified.

`SYMCLI_REPLICATE_MAX_BCV_SLEEP_TIME_FACTOR=<Factor>`

Provides a way to specify the maximum time that `symreplicate` sleeps before checking again to see if BCV devices have entered a specific state. The product of this value multiplied by the sleep time gives the maximum time that `symreplicate` sleeps. The factor is specified using a nonzero integer. If not specified, the default factor is 3.

By default, `symreplicate` sleeps between 10 and 30 seconds when checking on the state of BCV devices, up to a maximum time of 2 hours.

`SYMCLI_REPLICATE_MAX_RDF_SLEEP_TIME_FACTOR=<Factor>`

Provides a way to specify the maximum time that `symreplicate` sleeps before checking again to see if SRDF devices have entered a specific state. The product of this value multiplied by the sleep time gives the maximum time that `symreplicate` sleeps. The factor is specified using a nonzero integer.

If not specified, the default factor is 4.

Note: By default, `symreplicate` sleeps between 15 and 60 seconds when checking on the state of SRDF devices, up to a maximum time of 4 hours.

`SYMCLI_REPLICATE_TF_CLONE_EMULATION=<TRUE|FALSE>`

Indicates that TF/Clone emulation is enabled/disabled.

`FALSE` – (default) The TF/Clone emulation default is disabled.

`TRUE` - Indicates that clone emulation is enabled.

`SYMCLI_REPLICATE_PERSISTENT_LOCKS=<TRUE|FALSE>`

Allows device locks to persist in the event of a system crash or component failure.

`TRUE` - Causes symreplicate to acquire the device locks for the `symreplicate` session with the `SYMAPI_DLOCK_FLAG_PERSISTENT` attribute.

`FALSE` - The persistent attribute will not be used to acquire the device locks for the session. If the base daemon (storapi daemon) is running and persistent locks are not set, the base daemon will release the device locks in the event of a failure.

`SYMCLI_REPLICATE_CONS_SPLIT_RETRY=<NumRetries>`

Specifies the number of error recovery attempts that will be made when a consistent split operation fails because the timing window closed before the split operation completed. A default retry value of 3 will be used if the SYMCLI_REPLICATE_CONS_SPLIT_RETRY option parameter is not specified when a consistent split (`-consistent`) is requested. A retry value of 0 indicates that no retry attempts should be made.

`SYMCLI_REPLICATE_R1_BCV_EST_TYPE=<EstablishType>`

Specifies the establish type for the local/first hop BCV devices. *EstablishType* specifies the way that BCV establish operations will be executed by TimeFinder. One of the following values may be specified:

`SINGULAR` — BCV devices will be established one at a time; the next device will not be established until the previous device has been established.

`SERIAL` — BCV devices will be established as fast as the establish requests can be accepted by the Symmetrix array.

`PARALLEL` — BCV devices establish requests will be passed in parallel to each of the servicing DA directors.

`SYMCLI_REPLICATE_R1_BCV_DELAY=<EstablishDelay>`

Denotes how long to wait between issuing establish requests. Establish types of SINGULAR and PARALLEL, for an ‹*EstablishDelay*› can be specified through the SYMCLI_REPLICATE_R1_BCV_DELAY file parameter.

SYMCLI_REPLICATE_FINAL_BCV_EST_TYPE=<*EstablishType*>

Identifies the establish type for the remote/second hop BCV devices.

SYMCLI_REPLICATE_FINAL_BCV_DELAY=<*EstablishDelay*>

Indicates how long to wait between issuing establish requests for the remote/second hop BCV devices. For an establish type of PARALLEL the delay value indicates how long to wait before passing the next establish request to an individual servicing DA director. An establish delay of 0 to 30 seconds may be specified with a value of 0 being the default.

SYMCLI_REPLICATE_ENABLE_STATS=<*TRUE*|*FALSE*>

Enables or disables the gathering of statistics.

TRUE - (default) Indicates that statistics statistics gathering is enabled.

FALSE - Indicates that statistics gathering is to be disabled.

SYMCLI_REPLICATE_STATS_RESET_ON_RESTART=<*TRUE*|*FALSE*>

Resets statistics when a restart action is executed.

TRUE - Indicates that statistics are to be reset when restarting a symreplicate session.

FALSE - (default) Statistics are not reset upon restart of a symreplicate session.

Options file format

The options file should conform to the following syntax example, where the desired value is entered for the italicized text. Lines beginning with a "#" (comment) are ignored by SYMCLI:

```
#Comment
SYMCLI_REPLICATE_HOP_TYPE=<RepType>
SYMCLI_REPLICATE_CYCLE=<CycleTime>
SYMCLI_REPLICATE_CYCLE_OVERFLOW=<OvfMethod>
SYMCLI_REPLICATE_CYCLE_DELAY=<Delay>
SYMCLI_REPLICATE_NUM_CYCLES=<NumCycles>
SYMCLI_REPLICATE_USE_FINAL_BCV=<TRUE|FALSE>
SYMCLI_REPLICATE_LOG_STEP=<TRUE|FALSE>
SYMCLI_REPLICATE_GEN_TIME_LIMIT=<TimeLimit>
SYMCLI_REPLICATE_GEN_SLEEP_TIME=<SleepTime>
SYMCLI_REPLICATE_RDF_TIME_LIMIT=<TimeLimit>
SYMCLI_REPLICATE_RDF_SLEEP_TIME=<SleepTime>
SYMCLI_REPLICATE_BCV_TIME_LIMIT=<TimeLimit>
SYMCLI_REPLICATE_BCV_SLEEP_TIME=<SleepTime>
SYMCLI_REPLICATE_MAX_BCV_SLEEP_TIME_FACTOR=<Factor>
SYMCLI_REPLICATE_MAX_RDF_SLEEP_TIME_FACTOR=<Factor>
SYMCLI_REPLICATE_PROTECT_BCVS=<Protection>
SYMCLI_REPLICATE_TF_CLONE_EMULATION=<TRUE|FALSE>
SYMCLI_REPLICATE_PERSISTENT_LOCKS=<TRUE|FALSE>
SYMCLI_REPLICATE_CONS_SPLIT_RETRY=<NumRetries>
SYMCLI_REPLICATE_R1_BCV_EST_TYPE=<EstablishType>
SYMCLI_REPLICATE_R1_BCV_DELAY=<EstablishDelay>
SYMCLI_REPLICATE_FINAL_BCV_EST_TYPE=<EstablishType>
```

```
SYMCLI_REPLICATE_FINAL_BCV_DELAY=<EstablishDelay>
SYMCLI_REPLICATE_ENABLE_STATS=<TRUE|FALSE>
SYMCLI_REPLICATE_STATS_RESET_ON_RESTART=<TRUE|FALSE>
```

Note: For proper session behavior, either a *CycleTime* or a *Delay* time nonzero value should be specified, even though their default values are zero. The *RepType* must be specified.

Locked devices

Device external locks in the Symmetrix array are held during the entire `symreplicate` session, which is necessary to block other applications from altering device states while this session executes.

Note: Under certain circumstances, a `symreplicate` session may exit leaving the devices in a locked state. Device locks can be recovered, released or acquired to persist.

Recovering locks

If a `symreplicate` session terminates when an SRDF link goes down unexpectedly, the `symreplicate` session cannot restart after the SRDF link is brought back up, because of the locked devices. Use the `-recover` option with the `symreplicate start` or `restart` command to recover the device locks and restart the session.

Note: As long as the exact same devices are still locked under the lock holder ID of the previous `symreplicate` session, then the device locks can be recovered.

Releasing locks

You can optionally release the device external locks held in the Symmetrix array for a terminated SRDF/AR session. Locks may need to be released manually if a session is terminated unexpectedly due to a system crash or component failure. Device locks for a terminated session can be released manually for a device group, composite group or log file without restarting the session.

For example, to release devices locks on a terminated session for device group `prod` on Symmetrix `35002`, type:

```
symreplicate -g prod release -sid 35002
```

When the above command is executed, any device external locks associated with devices in device group `prod` that were locked from the previous SRDF/AR session that are still held will be released.

The following restrictions apply to releasing locks:

- The SRDF/AR session for the targeted devices must not be active.
- Devices must have been locked by the previous session and the lock holder ID must match the previous session's ID.
- The number of devices to be unlocked must be less than or equal to the total number of devices in the previous SRDF/AR session.

The force (`-force`) option is required to release device locks in the following situations:

- If the release action is requested in a clustered SRDF/AR environment on a host that did not initiate the session and the status of the session cannot be determined.
- If any of the devices' lock holder ID in the targeted SRDF/AR session do not match the session's lock hoder ID, and the user wants to release the devices locked with the session's lock holder ID.
- If the lock holder ID for some devices in the targeted SRDF/AR session do not match the lock holder ID of that session, and the user wants to release the devices locked with the session's original lock holder ID.

Acquiring persistent locks

If running the base daemon (SYMAPI daemon), device locks will automatically be released in the event of a system crash or component failure. Optionally, the device locks may be acquired using the persistent attribute by setting the `SYMCLI_REPLICATE_PERSISTENT_LOCKS` parameter to `TRUE` in the symreplicate options file. Refer to "Setting the symreplicate file parameters" on page 192 for additional information.

CHAPTER 9
SRDF Consistency Group Operations

This chapter describes how to create and maintain SRDF consistency groups using the SYMCLI. Topics include:

- Overview 200
- SRDF consistency group operations 202
- Dynamic modification of SRDF consistency groups 213
- SRDF consistency with a parallel database 225
- SRDF consistency with BCV access at the target site 226

Overview

An SRDF consistency group is a composite group comprised of Symmetrix SRDF devices enabled for remote database consistency.

The devices in the consistency group are configured to act in unison to maintain the integrity of a database when distributed across multiple Symmetrix arrays or across multiple devices within an array.

SRDF consistency protection software preserves the dependent-write consistency of devices within the group by monitoring data propagation from source devices to their corresponding target devices. If a source R1 device in the consistency group cannot propagate data to its corresponding R2 device, the SRDF consistency software suspends data propagation from all the R1 devices in the group.

This allows you to quickly recover from certain types of failures 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.

Note: You can also ues Domino modes to ensure the integrity of a remote database (refer to "Domino effect" on page 36).

Consistency protection using the SRDF daemon

The SRDF daemon (storrdfd) provides consistency protection for SRDF/A MSC and SRDF/S RDF-ECA consistency groups in multi-Symmetrix array environments as well as multiple SRDF groups within the same Symmetrix array.

For MSC consistency groups, the SRDF daemon performs cycle switching and cache recovery for all SRDF/A sessions within a consistency group. If a data flow interruption (such as a trip event) occurs, storrdfd

- Halts R1->R2 data propagation
- Analyzes the status of all SRDF/A sessions.
- Either commits the last cycle of data to the R2 targets or discards it.

For RDF-ECA consistency groups, storrdfd continuously polls SRDF/S sessions for data flow interruptions. If any R1 device is unable to propagate data to its R2 target, storrdfd suspends all R1->R2 data flow within an RDF-ECA consistency group. The storrdfd daemon ensures that you always have a consistent R2 copy of a database at the point in time in which a data interruption occurs.

Before the storrdfd daemon can monitor and manage a consistency group, you must create a composite group using the SRDF consistency option (`-rdf_consistency`) and then enable it using the `symcg enable` command. Refer to "Creating an SRDF consistency group" on page 203 for more information.

For operational examples of consistency protection, see Chapter 15, "Implementing Consistency Protection."

Enabling the SRDF daemon

Although the storrdfd daemon is optional, it is required for SRDF consistency group operations. By default, the storrdfd daemon is disabled and must be enabled for all applications using the SYMAPI configuration database file and SRDF consistency protection.

You enable storrdfd using the following SYMAPI options file setting:

```
SYMAPI_USE_RDFD=ENABLE
```

Each host running the SRDF daemon must also be running the base daemon (storapid).

The *EMC Solutions Enabler Symmetrix Command Reference Guide* explains how to start and stop daemons and explains other common daemon tasks.

Enabling the Group Naming Services daemon

The storrdfd daemon runs on each host for which SRDF consistency is required.

If the Group Naming Services (GNS) daemon is enabled, storrdfd relies on GNS to propagate updated CG definitions to all hosts locally attached to the same set of Symmetrix arrays.

If GNS is not enabled, you must manually recreate the updated CG definition on each one of these hosts.

You enable GNS on each host using the following SYMAPI options file setting:

```
SYMAPI_USE_GNS=ENABLE
```

IMPORTANT

When using GNS, enabling the gns_remote_mirror option in the daemon_options file will not mirror the CG if it includes any devices listed in "Mirroring exceptions" in th *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*. Refer to the Guide for a detailed description of GNS.

Redundant consistency protection

For redundant consistency protection of composite groups, you must simultaneously run two instances of the SRDF daemon on separate control hosts. Each control host must have a common view of the composite group being monitored, which can be accomplished by doing one of the following:

- Running the GNS daemon on each control hosts, as shown in Figure 31 on page 202, or
- Manually defining the composite group on all control hosts.

SRDF daemons running simultaneously on two different hosts perform independent monitoring and switching operations. If one fails, the other SRDF daemon takes it place, completing all pending tasks, such as committing the last cycle of data to the target site. By running redundant SRDF daemons, you avoid service interruptions caused by performance bottlenecks local to a control host, and link failures of the redundant SRDF daemons and the control hosts.

As shown in Figure 31 on page 202, Host-1 and Host-2 contain sets of the base daemon, the SRDF daemon, and the GNS daemon to ensure data consistency protection.

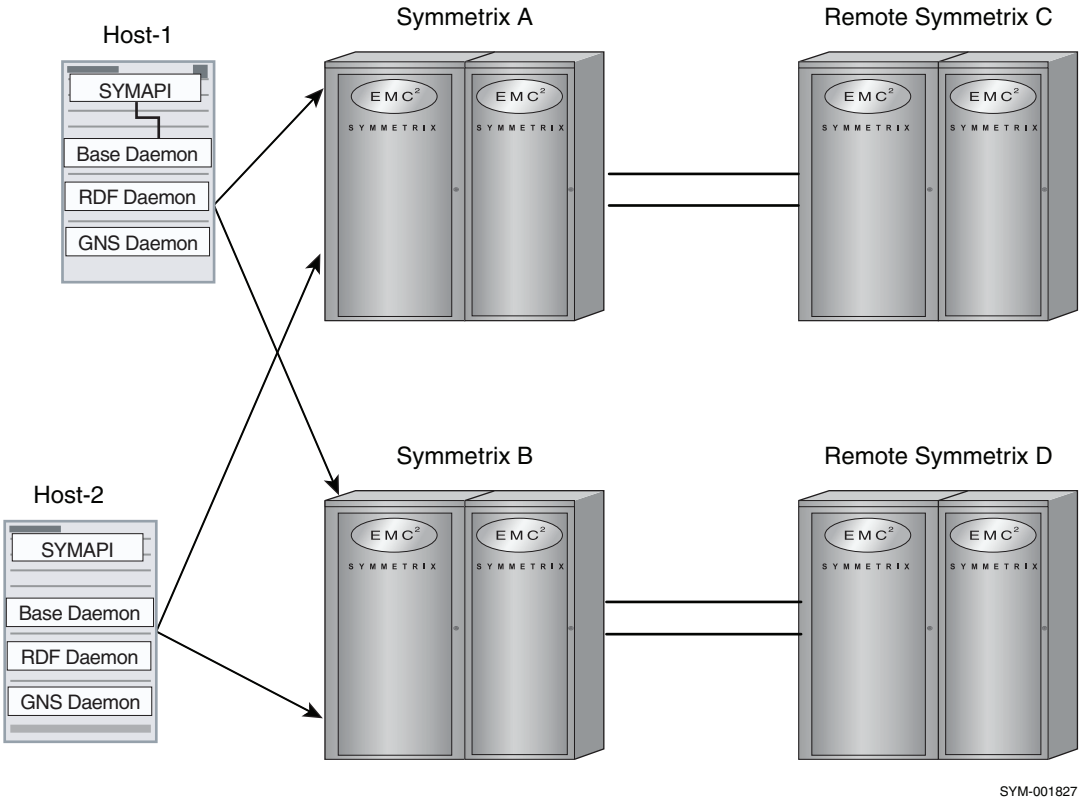


Figure 31 Running redundant hosts to ensure consistency protection

IMPORTANT

EMC strongly recommends running redundant SRDF daemons on at least two control hosts at each site. This ensures at least one SRDF daemon is available to perform time-critical, consistency monitoring operations.

EMC recommends that you do not run the SRDF daemon on the same control host running the database applications. However, you can use this control host to issue other control commands (such as SRDF, Timefinder, and Clone operations). As long as the control host is powerful enough to efficiently handle all CPU operations, and is configured with sufficient gatekeeper devices for all your management applications, you can also run ECC and Unisphere for VMAX along with the Solutions Enabler daemons shown in Figure 31.

SRDF consistency group operations

This section describes the control operations that you can perform on SRDF consistency groups.

Note: These operations require Symmetrix arrays running Enginuity 5671 and higher.

SRDF composite groups are initially created using the `symcg create` command and then populated with devices and device groups. For a composite group to be enabled as an SRDF consistency group, the group must be defined as a type RDF1, RDF2, or RDF21 and must be set with the consistency protection option (`-rdf_consistency`).

Note: A composite group type can change as a result of an `symrdf` control operation. For example, an RDF1 CG can change to an RDF2 CG when the device personalities are swapped. SRDF control operations (such as the `failover -establish` and `swap` operations) cannot change the type of an ANY composite group but can affect the devices in that CG.

For a list of control actions and the required SRDF pair states for consistency group operations, refer to "SRDF Pair State Reference" on page 497.

Note: Thin devices within a composite group for SRDF/S and SRDF/A configurations are supported on Symmetrix arrays running Enginuity 5773.150 and higher.

Creating an SRDF consistency group

The following steps illustrate how to build a consistency group when devices in the group are either all synchronous or all asynchronous. All devices containing application and system data must be included in the CG for each DBMS or across the DBMS controlling the multi-database transactions.

1. List all SRDF (RA) groups on the source Symmetrix arrays connected to the local hosts to determine which devices to include in the CG:

   ```
   symcfg list -rdfg all
   ```

2. Create a consistency group called `ConsisGrp` on one of the local hosts by specifying the SRDF type of the group and the `-rdf_consistency` option:

   ```
   symcg create ConsisGrp -type rdf1 -rdf_consistency
   ```

3. Add the devices from an SRDF (RA) group, such as `RDG 64`, into the `ConsisGrp`:

   ```
   symcg -cg ConsisGrp -sid 3264 addall dev -rdfg 64
   ```

4. In a database configuration with multiple local hosts, you must build the same consistency group on all local hosts in the configuration. You can use the `symcg export` command to manually transfer the CG definition, or if enabled, use GNS to automatically transfer it. The following commands create the `consisgrp.txt` text file containing the new `ConsisGrp` CG definition and then transfers it to `Host-1`:

   ```
   symcg export ConsisGrp -f consisgrp.txt
   rcp consisgrp.txt Host-1:/.
   ```

   The following `-rdf_consistency` option causes the imported `ConsisGrp` CG definition to be added to the SRDF consistency database on `Host-1`:

   ```
   symcg import ConsisGrp -f consisgrp.txt -rdf_consistency
   ```

5. Verify that all devices in the CG are either all synchronous or all asynchronous. For example, if the devices are currently operating with synchronous replication and you want them to be operating asynchronously, set the composite group for asynchronous replication:

```
symrdf -cg ConsisGrp set mode async
```

6. If the SRDF pairs are not in the Consistent or Synchronized state at this time (for example, the Split or Suspended state), you can initiate SRDF copying of R1 data to the R2 side. The device state is SyncInProg until the Consistent or Synchronized state is reached. With asynchronous replication, it may take two cycle switches for all devices to reach the Consistent state before the CG is consistent:

   ```
   symrdf -cg ConsisGrp establish
   ```

7. From one of the local hosts, enable the composite group for consistency protection (at which time the `ConsistGrp` CG becomes an SRDF consistency group managed by the SRDF daemon):

   ```
   symcg -cg ConsisGrp enable
   ```

   At this point, the SRDF daemon watches for any problems with R1->R2 data within the `ConsisGrp` CG.

Creating composite groups from various sources

This section explains how to create CGs from device groups, an RDBMS database, and a logical volume group.

Creating a composite group from an existing device group

You can translate the devices of an existing device group to a new or existing composite group. The following command translates and adds all devices from a device group named Symm64DevGrp to a composite group named `ConsisGrp`. The `-rdf_consistency` option adds the composite group to the SRDF consistency database on the host and makes the group capable of being enabled for SRDF consistency protection:

```
symdg dg2cg Symm64DevGrp ConsisGrp -rdf_consistency
```

Creating a composite group from an RDBMS database

You can also translate the devices of an existing RDBMS database or tablespace to a new or existing composite group. However, for SYMCLI to access a specified database, you must set the `SYMCLI_RDB_CONNECT` environment variable to the username and password of the system administrator's account. For example, when connecting locally, you can use the following command to set the variable to a username of "system" and a password of "manager." (The Bourne and Korn shells use the `export` command to set environment variables; the C shell uses the `setenv` command):

```
export SYMCLI_RDB_CONNECT=system/manager
```

When connecting by the network, you need to add a database-specific variable to the `RDB_CONNECT` definition. For example, connecting through the network in an Oracle environment means that you have an Oracle network listener process running. In this case, you need to add an Oracle connection string, such as the Transparent Network Substrate (TNS) alias name "api217", in the following command:

```
export SYMCLI_RDB_CONNECT=system/manager@api217
```

Similarly, connecting through the network in an SQL Server 2000 environment requires adding a string such as "HR" to indicate the ODBC data source administrator:

```
set SYMCLI_RDB_CONNECT=system/manager@HR
```

Optionally, you can set the `SYMCLI_RDB_TYPE` environmental variable to a specific type of database (for example, `oracle`, `informix`, `sqlserver`, or `ibmudb`) so that you do not have to include the `-type` option on the `symrdb rdb2cg` command line. The following command sets this variable to `oracle`:

```
export SYMCLI_RDB_TYPE=oracle
```

The following `symrdb rdb2cg` command translates the devices of an Oracle-type database named oradb to an RDF1 type composite group named `ConsisGrpDb`. The `-rdf_consistency` option adds the composite group to the SRDF consistency database on the host:

```
symrdb -type oracle -db oradb rdb2cg ConsisGrpDb -cgtype rdf1 -rdf_consistency
```

The following `symrdb tbs2cg` command translates the devices of an oracle type tablespace named orats to an RDF1 type composite group named `ConsisGrpTs`:

```
symrdb -type oracle -tbs orats tbs2cg ConsisGrpTs -cgtype rdf1 -rdf_consistency
```

Note: The E-Lab™ Interoperability Navigator which can be reached at http://elabnavigator.EMC.com provides detailed interoperability information.

With most RDBMS database systems, it is necessary to set up environment variables that are specific to that system. For example, Oracle systems use `ORACLE_HOME` and `ORACLE_SID`, and Sybase systems use `SYBASE` and `DSQUERY`. "Example 4: Creating a composite group from existing sources" on page 393 shows how to define the Oracle environment variables.

Creating a composite group from a logical volume group

You can also translate the devices of an existing logical volume group to a new or existing composite group using the `symvg` command. This command does not require setting up any environment variables before performing this operation.

To translate the devices of a logical volume group named LVM4vg to an RDF1 type composite group named `ConsisGrp`. The `-rdf_consistency` option adds the composite group to the SRDF consistency database on the host:

```
symvg vg2cg LVM4vg ConsisGrp -cgtype rdf1 -rdf_consistency
```

Note: The E-Lab™ Interoperability Navigator which can be reached at http://elabnavigator.EMC.com provides detailed interoperability information.

Enabling and disabling SRDF consistency protection

You can enable or disable consistency protection for all the devices in a composite group. When you enable the composite group for consistency, the group is referred to as an SRDF consistency group.

Note: When consistency protection is enabled on a device, you cannot have domino effect mode turned on and vice versa.

When a composite group is enabled for consistency protection:

- Its name cannot be changed without first disabling the consistency protection. You can then re-enable it using the new name of the composite group.
- If the composite group is enabled for SRDF/A consistency protection, then the SRDF daemon will begin performing cycle switches on the SRDF groups within the composite group (or named subset in MSC mode). This means that the cycle switches for all SRDF groups will be performed at the same time. The interval between these cycle switches is determined by the smallest minimum cycle time defined on the R1 SRDF groups that are part of the composite group (or named subset). The smallest minimum cycle time supported by the SRDF daemon is 3 seconds, so this value is used if the smallest minimum cycle time across all component groups is less than 3 seconds.
- If you change the minimum cycle time for any of the R1 SRDF groups while the composite group (or named subset) is enabled for SRDF/A consistency protection, the new minimum cycle time will not take effect until you disable consistency protection and then re-enable it.
- If you change the minimum cycle time, the new minimum cycle time will not take effect until you disable consistency protection and then re-enable it.
- You can change contents of a composite group by doing one of the following:
  - Disable consistency protection on a composite group while you add or remove devices, and then re-enable consistency protection after editing the composite group. However, this leaves the devices in the composite group unprotected during the time required to edit and then re-enable the composite group.
  - For an RDF1 composite group, you can dynamically modify the composite group while maintaining consistency protection during the editing process. For more information, refer to "Dynamic modification of SRDF consistency groups" on page 213.

Composite group level vs. SRDF group name level

You can enable and disable consistency protection at the composite group level or at the SRDF group name level.

Composite group level

By enabling a composite group for consistency protection, all devices within the CG operate as a single unit. Therefore, if one R1 device in a CG is unable to propagate data to its R2 target, the SRDF daemon suspends the SRDF links of all the devices within that CG. This is referred to as enabling the CG at the composite group level. To enable consistency protection at the CG level, all device mirrors must be operating in the same SRDF mode (in other words, all device mirrors must be operating synchronously or asynchronously).

To enable consistency protection at the CG level, encompassing all device pairs in the `prod` CG, type:

```
symcg -cg prod enable
```

To disable consistency protection for all device pairs in the enabled `prod` CG, type:

```
symcg -cg prod disable
```

SRDF group name level

You can also enable consistency protection to a named subset of devices within a CG. When an R1 device in a CG cannot send data to its target, the SRDF daemon suspends the SRDF links for only those devices in the subset of the CG. This subset is based on the SRDF group names of a CG and is referred to as enabling the CG at the SRDF group name level. This type of protection is particularly useful for concurrent devices that have one mirror operating in synchronous mode and the other mirror operating in asynchronous mode.

To enable consistency protection at the SRDF group name level, you must first define one or more named subsets of devices within the composite group. A subset can consist of one or more of the SRDF groups within the composite group.

For example, the CG SALES consists of a set of concurrent SRDF devices distributed across two Symmetrix arrays, 000195900076 and 000195900077.

On array 000195900076:

- SRDF group 100 is operating in asynchronous mode, and
- SRDF group 120 is operating in synchronous mode.

On array 000195900077:

- SRDF group 101 is operating in asynchronous mode, and
- SRDF group 121 is operating in synchronous mode.

The following commands create two named subsets of the composite group, one containing the asynchronous SRDF groups and the other containing the synchronous groups:

```
symcg -cg SALES set -name sales1 -rdfg 76:100
symcg -cg SALES set -name sales1 -rdfg 77:101


symcg -cg SALES set -name sales2 -rdfg 76:120
symcg -cg SALES set -name sales2 -rdfg 77:121
```

The two named subsets can then be independently enabled for consistency protection:

```
symcg -cg SALES enable -rdfg name:sales1
symcg -cg SALES enable -rdfg name:sales2
```

After a subset of a CG is enabled for consistency protection at the SRDF group name level:

- You cannot change the name of that subset without first disabling it.
- You cannot modify the contents of that subset, by adding or removing SRDF groups, without first disabling its consistency protection.
- For an RDF1 composite group, you can dynamically modify the contents of a subset while maintaining consistency protection during the editing process. For more information, refer to "Dynamic modification of SRDF consistency groups" on page 213. Otherwise you must first disable consistency protection for the SRDF group name while adding or removing devices, and then re-enable consistency protection for that SRDF group name.

- Enabling a composite group at the CG level and SRDF group name level are mutually exclusive. If a composite group is enabled at the CG level, no part of it can be simultaneously enabled at the SRDF group name level. Conversely, if a subset of the group is enabled at the SRDF group name level, the group cannot be enabled at the CG level.

SRDF/S devices

The `enable` action enables consistency protection either across all synchronous-mode devices in a consistency group, or across all synchronous-mode devices in a named subset of a composite group. If any R1 devices in an SRDF/S consistency group cannot propagate data to their corresponding R2 targets, the SRDF daemon suspends data propagation from all R1 devices in the consistency group, halting all data flow to the R2 targets.

To enable consistency protection for SRDF/S pairs in the `prod` CG, type:

```
symcg -cg prod enable
```

To disable consistency protection for SRDF/S pairs in the `prod` CG, type:

```
symcg -cg prod disable
```

SRDF/A devices

The `enable` action enables consistency protection either across all asynchronous-mode devices in a consistency group, or across all asynchronous-mode devices in a named subset of a composite group. If an SRDF/A session that was enabled for consistency protection cannot propagate data from the R1 devices to their corresponding R2 target, Enginuity deactivates that session, suspending data propagation for all devices in the SRDF/A session and preserving R2 consistency. If the consistency group or named subset of a composite group is comprised of multiple SRDF/A sessions, the SRDF daemon then suspends data propagation for the other SRDF/A sessions, halting all data flow to the R2 targets in order to preserve R2 consistency.

To enable consistency protection for SRDF/A pairs in the `prod2` CG, type:

```
symcg -cg prod2 enable
```

To disable consistency protection for SRDF/A pairs in the `prod2` CG, type:

```
symcg -cg prod2 disable
```

Enabling SRDF consistency protection for concurrent SRDF devices

For a concurrent SRDF configuration, you can enable consistency protection at the composite group level or at the SRDF group name level.

Composite group level

When both mirrors of the concurrent R1 devices in the composite group are operating in synchronous mode or asynchronous mode, you can enable SRDF consistency protection at the composite group level.

Enabling SRDF consistency protection at the composite group level causes the SRDF daemon to monitor the SRDF groups that represent the two mirrors together. If the two groups are operating in asynchronous mode, they cycle-switch together. In either mode, if a concurrent R1 device is unable to propagate its data to either of its remote R2 partners, the SRDF daemon suspends the SRDF links for both groups to preserve consistency of the R2 data.

For example, the composite group `prod` contains a concurrent R1 with two asynchronous target mirrors. The following command enables these two target mirrors together with MSC consistency protection.

```
symcg -cg prod enable
```

In the example above, if the composite group `prod` contained a concurrent R1 with two synchronous mirrors, they would be enabled with RDF-ECA consistency protection.

Group name level

If the two mirrors of the concurrent R1 devices in the composite group are operating in different modes (one mirror in synchronous mode and the other mirror in asynchronous mode), SRDF consistency protection cannot be enabled at the composite group level. Instead, you must individually enable each group representing the device mirrors by its group name.

Enabling SRDF consistency at the group name causes the SRDF daemon to monitor the SRDF groups separately so that if a concurrent R1 device is unable to propagate its data to one of its remote R2 partners, the daemon suspends the SRDF links for only the group representing that R2 mirror.

Use the `symcg` command to define the name `cGrpA` to represent SRDF group 55 on Symmetrix 123:

```
symcg -cg prod set -name cGrpA -rdfg 123:55
```

Then use the `symcg` command to enable consistency protection for that SRDF group:

```
symcg -cg prod enable -rdfg name:cGrpA
```

If the mirrors in SRDF group 55 are operating in asynchronous mode, the SRDF group is enabled with MSC consistency protection; if they are operating in synchronous mode, the SRDF group is enabled with RDF-ECA protection.

The same steps can be followed to enable the other SRDF group for consistency protection, defining a different name for that group.

Table 16 lists the combinations of consistency protection modes allowed for the mirrors of a concurrent relationship.

Table 16 Supported consistency modes for concurrent mirrors

R1->R2 (first mirror)	R1->R2 (second mirror)
MSC	None
MSC	RDF-ECA
MSC	MSC
RDF-ECA	None
RDF-ECA	RDF-ECA
RDF-ECA	MSC
None	None
None	MSC
None	RDF-ECA

Note: Enabling both mirrors of a concurrent R1 for RDF-ECA requires Symmetrix arrays running Enginuity 5874 or higher, and enabling both mirrors of a concurrent R1 for MSC requires Symmetrix arrays running Enginuity 5875 or higher.

Enabling SRDF consistency protection for cascaded SRDF devices

You can enable devices in a cascaded SRDF relationship for SRDF consistency for the first hop (R1->R21) and the second hop (R21->R2).

You can enable SRDF consistency protection for cascaded devices using a composite group or an SRDF group name, but not both. For example, you *cannot* enable the first hop using a composite group and then enable the second hop using its SRDF group name.

To enable consistency protection for the second hop, type:

```
symcg -cg prod -rdfg name:sitez -hop2 enable
```

Table 17 lists the consistency protection allowed for cascaded SRDF device pairs.

Table 17 Consistency protection allowed for cascaded SRDF pairs

R1->R21	R21->R2
RDF-ECA	None
MSC	None
RDF-ECA	MSC
None	MSC

Table 18 lists the consistency protection allowed for diskless cascaded SRDF pairs. For more information, refer to "Managing a diskless cascaded environment" on page 164.

Table 18 Consistency protection allowed for diskless cascaded SRDF pairs

R1->R21	R21->R2
RDF-ECA	None
RDF-ECA	MSC
None	MSC

Checking if device pairs are enabled for consistency protection

The `symrdf verify -enabled` command validates whether device pairs are enabled for consistency protection. You can also check the pair state and SRDF mode using this command, as shown in the last example.

Use the following command to verify whether the device pairs in the STAGING group are enabled for consistency protection:

```
symrdf -g STAGING verify -enabled
```

If none of the device pairs in the STAGING group are enabled for consistency protection, the following message displays:

```
None of the devices in the group 'STAGING' are 'Enabled'.
```

However, if all devices in the STAGING group were enabled for consistency protection, the following message displays:

```
All devices in the group 'STAGING' are 'Enabled'.
```

Use the following command to verify whether the device pairs in the STAGING group are enabled for consistency protection and are in the synchronized OR consistent pair state:

```
symrdf -g STAGING verify -enabled -synchronized -consistent
```

If all devices are enabled and in the synchronized OR consistent pair state, the following message displays:

```
"All devices in the group 'STAGING' are 'Enabled' and in
'Synchronized, Consistent' states."
'Synchronized, Consistent' states."Blocking symcg enable on R2 side
```

Blocking symcg enable on R2 side

You can execute the `symcg enable` command from the R1 or R2 side of an SRDF relationship. To prevent the `symcg enable` operation from being executed on the R2 side, Solutions Enabler provides an option called `SYMAPI_ALLOW_CG_ENABLE_FROM_R2` in the options file that you can `ENABLE` (the default) or `DISABLE`. If set to disable, Solutions Enabler blocks the composite group from being enabled on the R2 side. This option prevents a situation where the SRDF daemon is running on the R2 side but cannot communicate with the R1 side to close the RDF-ECA window due to a link failure.

Deleting an SRDF consistency group

When you delete an SRDF consistency group from a CG, the SRDF daemon stops monitoring the CG. As a result, SRDF consistency protection on the R2 data cannot be guaranteed even though the devices formerly in the CG may remain enabled. To eliminate any confusion, always disable consistency protection on a group before deleting it, as explained in "Enabling and disabling SRDF consistency protection" on page 205. If there are members in the group, the command is rejected unless you use the force (-force) option.

For example, to delete the SRDF consistency group `mycg1` (with members) after disabling its consistency protection, type:

```
symcg delete mycg1 -force
```

Note: If you have not disabled consistency protection on a group before deleting it, you must use the `-symforce` option in the above command. The composite group remains enabled but is removed from the SYMAPI database.

Suspending SRDF consistency protection

In an SRDF consistency group where all devices are either synchronous or asynchronous, you can suspend consistency protection for all devices by issuing a `suspend`, `split` or `failover` command to the SRDF consistency group. This is also known as manually tripping the group.

The difference between `symrdf -cg suspend` and `symrdf -cg split` is the state of the R2 devices at the end of the deactivation:

- With `suspend`, the R2 devices are in the write disabled state and cannot be accessed by the target-side hosts, thus maintaining the consistency of the R2 database copy with the production copy on the R1 side.
- With `split`, the R2 devices are enabled for both reads and writes by the target-side hosts.

When the same consistency group is defined on multiple hosts, you can initiate a suspend operation from any host provided the consistency group is enabled. The following command deactivates consistency in a consistency group named `ConsisGrp`. The `-force` option is required here (and with `split` or `failover`) to ensure that you really want to stop the SRDF mirroring operation and suspend consistency protection:

```
symrdf -cg ConsisGrp suspend -force
```

To resume the SRDF links between the SRDF pairs in the SRDF consistency group and I/O traffic between the R1 devices and their paired R2 devices, use the `symrdf -cg resume` command:

```
symrdf -cg ConsisGrp resume
```

Consistency protection is automatically restored upon resumption of the link. Consistency protection is never disabled unless you specifically perform the `symcg -cg disable` operation.

For asynchronous replication, you can use the `symrdf -cg verify` command while including the `-cg_consistent` option to ensure that the SRDF consistency group is SRDF-consistency enabled and in a consistent state. This means that at least two cycle switches have occurred since all devices in each SRDF (RA) group reached a consistent state:

```
symrdf -cg ConsisGrp verify -cg_consistent
```

For synchronous replication, verify using the `-synchronized` option:

```
symrdf -cg ConsisGrp verify -synchronized
```

Note: If you execute the `failover` command on both mirrors of a concurrent R1 device, it converts the concurrent R1 into a concurrent R2 with a restore on both mirrors of that concurrent R2.

Using the msc_cleanup command

The SRDF daemon automatically performs MSC cleanup operations for devices in an MSC-enabled SRDF/A session during the processing of any SRDF control operation.

MSC cleanup operations either:

- Discard any incomplete SRDF/A data, or
- Commit completed data to the R2 to maintain dependent write consistency.

The MSC cleanup operation can be performed manually by executing the `symrdf msc_cleanup` command for a composite group at the R1 or R2 site, or by SRDF group at the R2 site.

For instance, if a failure (such as a link failure) causes protection to be triggered, the daemon may not be able to process all cleanup operations for the R2 devices where the receive and apply delta sets reside. In this case, the `symrdf msc_cleanup` command can be executed manually from the R2 site. If a consistency group definition is unavailable at the R2 site, the cleanup operation can be performed by directing the command to an SRDF (RA) group that was included as part of the consistency group.

For example, to perform cleanup operations from the remote host at the R2 site for Symmetrix 1123 and direct the command to SRDF group 4, type:

```
symrdf -sid 123 -rdfg 4 msc_cleanup
```

To check whether a MSC cleanup operation is required, use the `symcfg list` command with the `-rdfg all` option to display a list of SRDF (RA) groups on a specified Symmetrix array (`-sid`). This command displays flag information for SRDF groups operating in SRDF/A mode. The RDFA "Flags M" column denotes whether an MSC cleanup operation is required.

Dynamic modification of SRDF consistency groups

When the CG is enabled at the composite group level or the SRDF group name level, you can dynamically add or remove devices from an RDF1 CG while retaining its consistency protection for the following device types:

- Simple R1
- Concurrent R11
- Cascaded R1

The `symcg modify` command with the `add` and `remove` options performs dynamic modification of SRDF consistency groups.

For comprehensive command examples with corresponding sample outputs of this feature, see "Example 6: Dynamic modification of SRDF consistency groups" on page 404.

SRDF daemon interaction

Review the following to understand how the SRDF daemon maintains consistency protection of a CG during dynamic modification:

- The SRDF daemon must be running locally on the host where the dynamic modification operation is issued.
- The SRDF daemon on the local host continuously monitors the consistency group that is being changed.
- The SRDF daemons running on other hosts, do the following:
  - On hosts running GNS, SRDF daemons monitor the consistency group while it is being modified as long as these hosts are locally attached to the same set of Symmetrix arrays as the control host. Depending on the timing of the GNS updates, there may be a brief period during which the SRDF daemon stops monitoring the consistency group while waiting for the updated consistency group definition to propagate to the local GNS daemon.

- On hosts *not* running GNS, SRDF daemons running Solutions Enabler versions lower than 7.3.1 stop monitoring the CG during dynamic modification. To restart the monitoring of the modified consistency group on these hosts, perform the following steps after the dynamic modification operation completes:
  - Export the new definition to a file using the `symcg export` operation on the control host.
  - Send the file created by the `symcg export` operation to the other host.
  - Import the new definition file using the `symcg import` operation on the control host.

IMPORTANT

EMC strongly recommends running GNS on your hosts to ensure consistency protection while dynamically modifying CGs.

Command restrictions

The following guidelines apply to both the `add` and `remove` options of the `symcg modify` command:

- The `symcg modify` command is only allowed if all the local and remote Symmetrix arrays of the CG are reachable and running Enginuity version 5773 or higher.
- The SRDF daemon must be running locally on the host where the `symcg modify` command is issued.
- The `symcg modify` command only applies to RDF1 composite groups. It is not allowed for RDF2, RDF21, or type=ANY composite groups.
- The `symcg modify` command is not allowed for CGs consisting of device groups.
- The `symcg modify` command is not allowed for CGs containing both concurrent and cascaded SRDF devices.
- The `symcg modify` command is not allowed if any of the devices in the CG are in STAR mode. However, the `symstar modifycg` command lets you modify such devices, as explained in the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*.
- The SRDF groups affected by the `symcg modify` command cannot contain any devices enabled for consistency protection by another CG.
- Devices within SRDF groups of the CG to be modified must be in one of the following SRDF pair states:
  - Synchronized
  - SyncInProg with invalid tracks owed to the R2
  - Consistent with no invalid tracks
  - Within an affected SRDF group, device pairs can be a mixture of Synchronized and SyncInProg or a mixture of Consistent and SyncInProg.

Note: In the event the `symcg modify` command fails, you can rerun the command or issue `symcg modify -recover`. No control operations are allowed on a CG until after a recover completes on that CG.

Preparing the staging area

Before dynamic modification of SRDF consistency groups can take place, you must create a staging area that mirrors the configuration of the CG. The staging area consists of the following:

- SRDF groups containing the device pairs to be added to a consistency group using the `symcg modify -add` command.
- SRDF groups for receiving the device pairs removed from a consistency group using the `symcg modify -remove` command.

The SRDF groups in the staging area must be established between the same Symmetrix arrays as the SRDF groups in the consistency group. In the case of a concurrent CG and a cascaded CG, the SRDF groups in the staging area are established among three Symmetrix arrays.

Restrictions

The following restrictions apply to the SRDF groups and devices in the staging area:

- SRDF groups cannot be part of an SRDF/Star configuration.
- Devices cannot be enabled for consistency protection.
- Devices cannot be defined with SRDF/Star SDDF (Symmetrix Differential Data Facility) sessions.
- BCVs are not allowed.
- All devices must be SRDF dynamic and of the same type:
  - Simple R1 devices
  - Concurrent R11 devices
  - Cascaded R1 devices with diskless R21 devices or cascaded R1 devices with non-diskless R21 devices
- All device pairs must set in the same mode:
  - Adaptive copy disk
  - Adaptive copy write pending for diskless R21->R2 device pairs

Dynamic add operation

The dynamic modify add operation moves device pairs from the staging area into the SRDF groups of a consistency group. To perform this operation, devices in the staging area must be in one of the following SRDF pair states for each SRDF group:

- Synchronized
- SyncInProg with invalid tracks owed to the R2
- Suspended

- Suspended with invalid tracks owed to the R2

If any device pair is Suspended (with or without invalid tracks) on any of its SRDF groups, then the device pairs in the same SRDF group must all be Suspended.

Example: Adding devices to the R1CG consistency group

Figure 32 shows a staging area for an R1->R2 configuration whose RDFG 101 is established between the same Symmetrix array as the RDFG 100 in the R1CG consistency group.

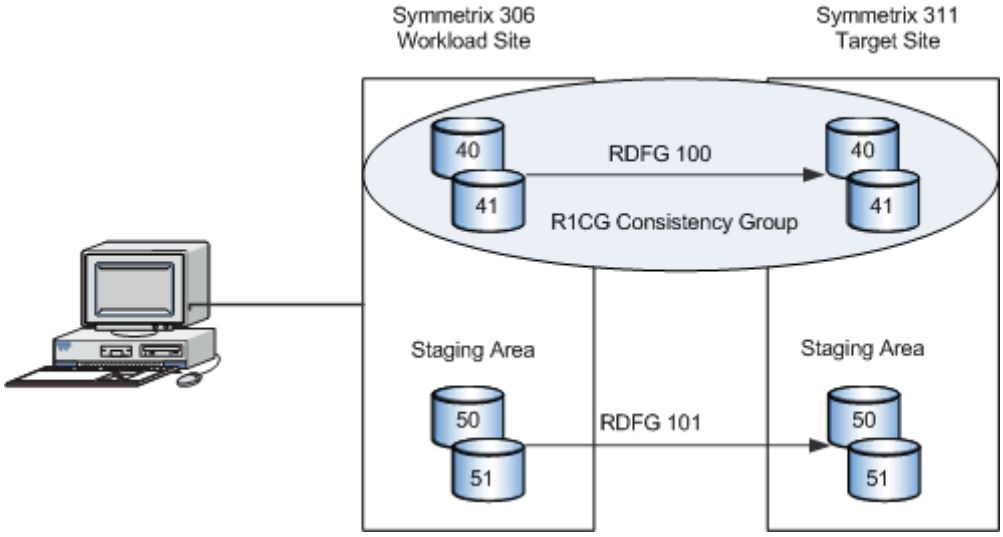


Figure 32 Preparing the staging area for adding devices to the R1CG consistency group

Figure 33 shows the R1CG consistency group after the dynamic modify add operation has completed. Since devices 50 and 51 were moved to R1CG, the staging area contains the empty RDFG 101.

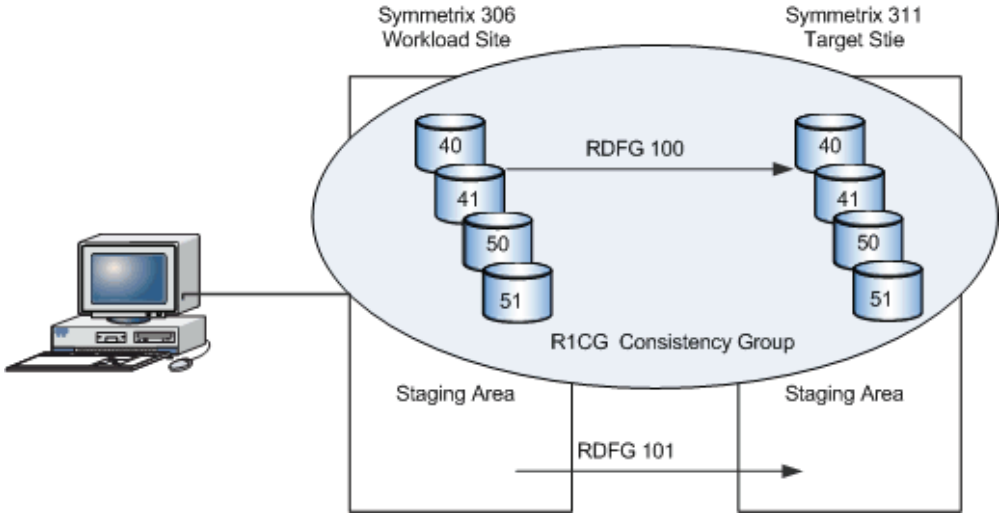


Figure 33 R1CG consistency group after a dynamic modify add operation

Dynamic remove operation

The dynamic modify remove operation moves the device pairs from the consistency group into the SRDF groups in the staging areas. To prepare the staging area for this operation, create the SRDF groups for receiving the device pairs removed from a consistency group.

Note: The dynamic modify remove operation must never leave an SRDF group empty.

Example: Removing devices from the MyR1 consistency group

Figure 34 shows a staging area for an R1->R2 configuration whose RDFG 34 is established between the same Symmetrix array as RDFG 32 in the MyR1 consistency group. In this example, RDFG 34 was configured to receive devices removed from RDFG 32.

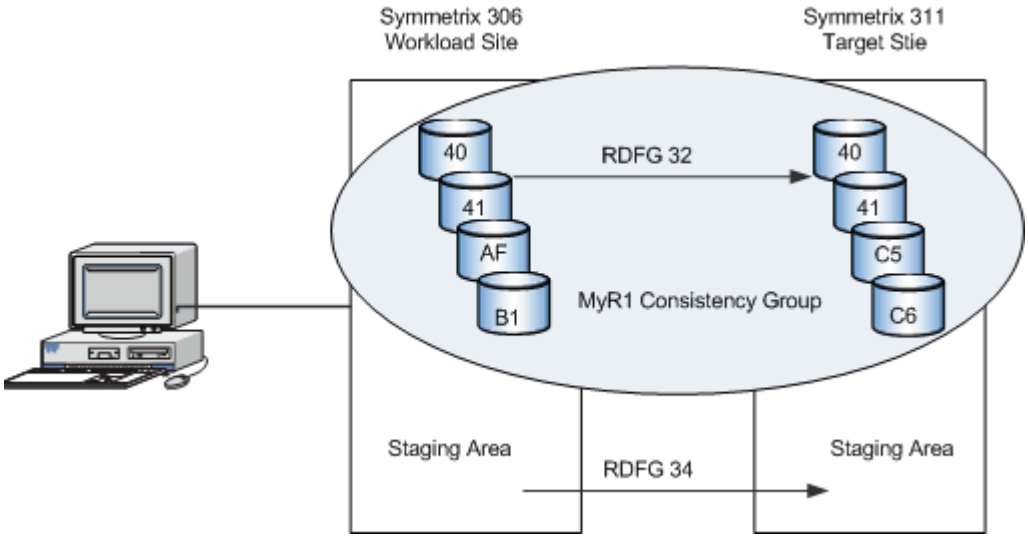


Figure 34 Preparing the staging area for removing devices from the MyR1 CG

Figure 35 shows the MyR1 consistency group and its staging area after the dynamic modify remove operation has completed.

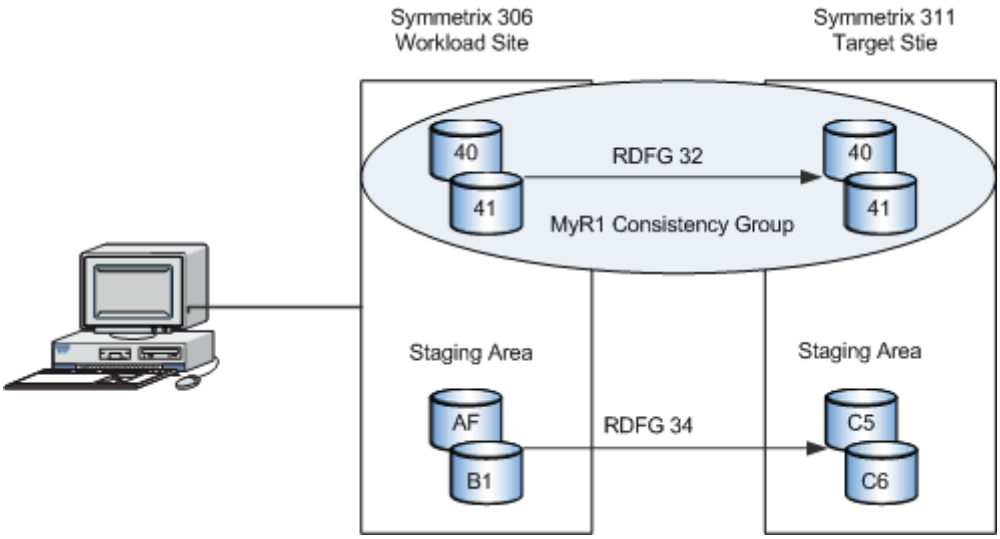


Figure 35 MyR1 CG after a dynamic modify remove operation

Dynamically adding devices

This section explains how to use the `symcg modify -add` command.

Restrictions

The following are restrictions for dynamically adding devices to an SRDF consistency group using the `symcg modify -add` command:

- The `symcg modify -add` command:
  - Cannot add new SRDF groups to the CG.
  - Cannot add a concurrent R11 device to a CG enabled at the composite group level.
  - Pprohibits adding both mirrors of a concurrent R11 device to the same SRDF group name.
  - Cannot add a triangle of devices to a CG. In other words, a concurrent R11 device cannot have one R1 mirror paired with an R21 device, which is then paired with an R22 device that is paired with the other R1 mirror of the concurrent R1 device.
  - Prohibits adding a cascaded R1 device to a concurrent CG.
  - Prohibits adding a concurrent R1 device to a cascaded CG.
- If the target is a cascaded CG, the operation must be enabled by CG hop 1 or by the SRDF group name hop 1.
- If the target is a cascaded CG and the devices to be added are simple R1 devices, the CG cannot be enabled by CG hop 2 or by SRDF group name hop 2.
- If the target is a cascaded CG and the devices to be added are cascaded R1 devices paired with diskless R21 devices, then all R21 devices in the affected SRDF group must also be diskless.
- If the target is a cascaded CG and the devices to be added are cascaded R1 devices paired with non-diskless R21 devices, then all R21 devices in the affected SRDF group must be non-diskless.

Adding devices to an RDF1 consistency group

Table 19 shows the allowable device types for a dynamic modify add operation on a composite group enabled for consistency protection at the composite group level and the SRDF group name level. This RDF1 CG is not concurrent or cascaded.

Table 19 Allowable device types for adding devices to an RDF1 CG

Device type in staging area	Enabled at CG level	Enabled at SRDF group name level
Simple R1 (R1->R2)	Allowed	Allowed
Concurrent R11	Not allowed	Only allowed if both affected SRDF groups in the CG already exist and are assigned to different SRDF group names
Cascaded R1	Not allowed	Not allowed

Example

To move devices 50 and 51 from SRDF group 101 in the staging area to SRDF group 100 in R1CG on Symmetrix 306, type:

```
symcg -cg R1CG modify -add -sid 306 -stg_rdfg 101 -devs 50:51 -cg_rdfg 100
```

To check if the devices were added to R1CG, type:

```
symrdf -cg R1CG query -detail
```

Adding devices to a concurrent RDF1 consistency group

Before performing this procedure, review the information in "Enabling SRDF consistency protection for concurrent SRDF devices" on page 208.

Table 20 shows the allowable device types for a dynamic modify add operation on a concurrent RDF1 composite group enabled for consistency protection at the composite group level and the SRDF group name level.

Table 20 Allowable device types for adding devices to a concurrent RDF1 CG

Device type in staging area	Enabled at CG level	Enabled at SRDF group name level
Simple R1 (R1->R2)	Allowed	Allowed
Concurrent R11	Not allowed	Only allowed if each mirror is assigned to a different SRDF group
Cascaded R1	Not allowed	Not allowed

Table 21 lists the allowable consistency modes for the SRDF groups of a concurrent CG.

Table 21 Supported consistency modes for concurrent SRDF groups

SRDF group 1 (first mirror)	SRDF group 2 (second mirror)
RDF-ECA	RDF-ECA
RDF-ECA	MSC
RDF-ECA	Not enabled
Not enabled	RDF-ECA
MSC	RDF-ECA
MSC	MSC
MSC	Not enabled
Not enabled	MSC

Example

Figure 36 on page 220 shows a concurrent SRDF configuration sharing the staging area among Symmetrix 306, 311, and 402. The staging area contains devices 20 and 21. This example adds device 20 to two independently-enabled SRDF groups of a CG.

Since the SRDF groups 70 and 71 of ConCG operate in different SRDF modes, they were enabled independently for consistency protection using the following SRDF group names:

- *Boston*: device pairs operate in SRDF/S mode and are set for RDF-ECA consistency protection.
- *New York*: device pairs operate in SRDF/A mode and are enabled for MSC consistency protection

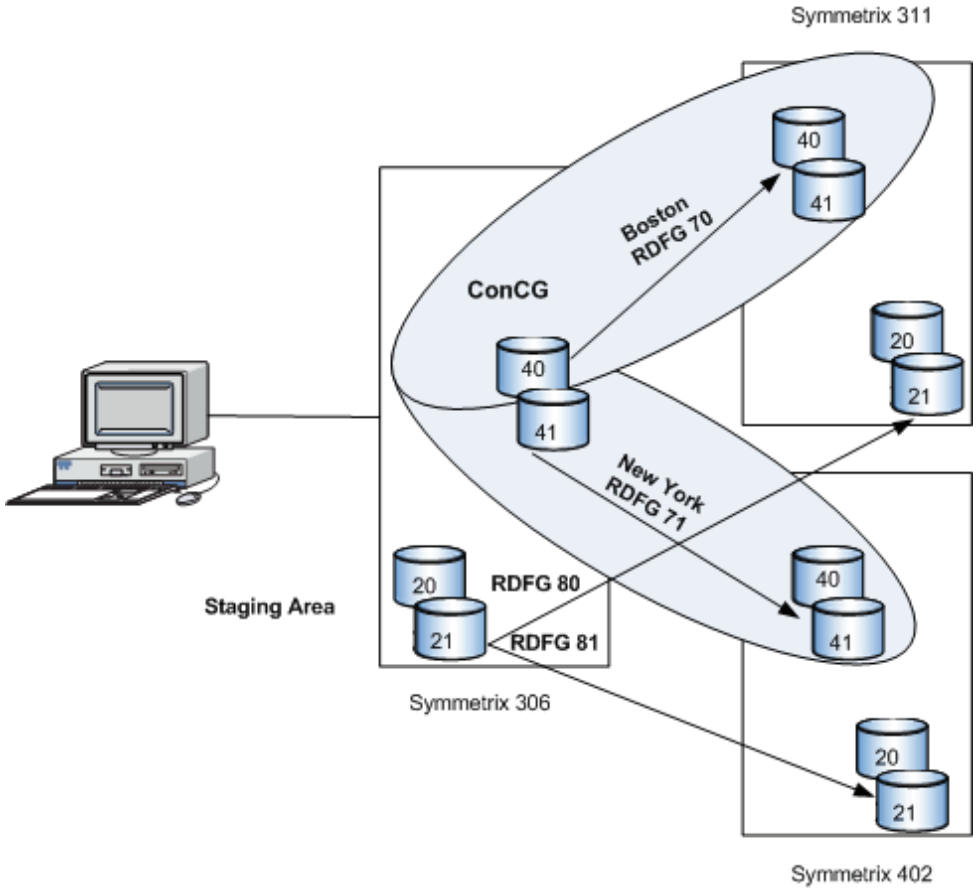


Figure 36 Adding a device to independently-enabled SRDF groups of a concurrent CG

To add only device 20 from the staging area into SRDF groups 70 (Boston) and 71 (New York) of `ConCG`, type:

```
symcg -cg ConCG modify -add -sid 306 -stg_rdfg 80,81 -devs 20 -cg_rdfg 70,71
```

To check if the devices were added to `ConCG`, type:

```
symrdf -cg ConCG query -detail
```

Adding devices to a cascaded RDF1 consistency group

Before performing this procedure, review the information in "Enabling SRDF consistency protection for cascaded SRDF devices" on page 210.

Table 22 shows the allowable device types for a dynamic modify add operation on a cascaded R1 composite group enabled for consistency protection at the composite group level and the SRDF group name level.

Table 22 Allowable device types for adding devices to a cascaded RDF1 CG

Device type in staging area	Enabled at CG level			Enabled at SRDF group name level		
	Hop 1 enabled Hop 2 not enabled	Hop 1 enabled Hop 2 enabled	Hop 1 not enabled Hop 2 enabled	Hop 1 enabled Hop 2 not enabled	Hop 1 enabled Hop 2 enabled	Hop 1 not enabled Hop 2 enabled
Simple R1 (R1->R2)	Allowed	Not allowed	Not allowed	Allowed	Not allowed	Not allowed
Concurrent R11	Not allowed	Not allowed	Not allowed	Not allowed	Not allowed	Not allowed
Cascaded R1	Allowed	Allowed	Not allowed	Allowed	Allowed	Not allowed

Table 23 lists the allowable consistency modes for the hops of a cascaded CG.

Table 23 Supported consistency modes for cascaded hops

R1->R21 (hop 1)	R21->R2 (hop 2)
RDF-ECA	MSC
RDF-ECA	Not enabled
MSC	Not enabled

Example

Figure 37 shows a cascaded SRDF configuration sharing the staging area among Symmetrix 306, 311, and 402. The staging area contains devices 20 and 21 to be added to `CasCG`.

The hops were independently enabled for consistency protection using the following SRDF group names:

- *New York*: device pairs operate in SRDF/S mode and are set for RDF-ECA consistency protection.

- *New Jersey*: device pairs operate in SRDF/A mode and are enabled for MSC consistency protection

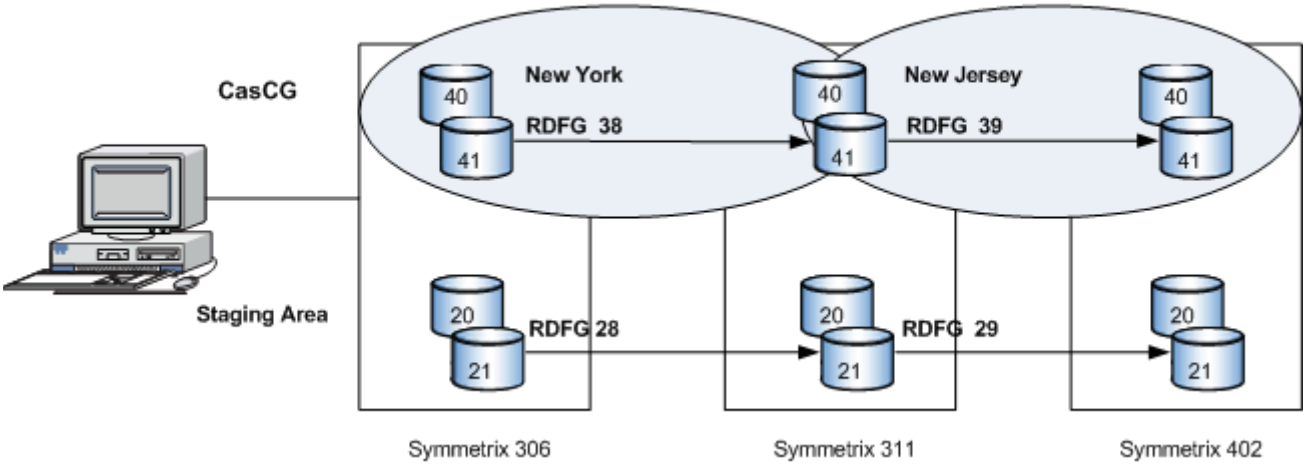


Figure 37 Adding devices to independently-enabled SRDF groups of a cascaded CG

To add devices 20 and 21 from the staging area into SRDF groups 38 (New York) and 39 (New Jersey) of `CasCG`, type:

```
symcg -cg CasCG modify -add -sid 306 -stg_rdfg 28 -devs 20:21 -stg_r21_rdfg 29
   -cg_rdfg 38 -cg_r21_rdfg 39
```

To check if the devices were added to `CasCG`, type:

```
symrdf -cg CasCG query -detail -hop2
```

Dynamically removing devices

This section explains how to use the `symcg modify -remove` command.

The dynamic modify remove operation moves the device pairs from the consistency group and places them into the staging area. These device pairs are left in adaptive copy disk mode or adaptive copy write pending mode (if the pair is a diskless R21->R2) and are in a Suspended state.

Note: The dynamic modify remove operation must never leave an SRDF group empty.

Restrictions

The following are restrictions for dynamically removing devices from an SRDF consistency group using the `symcg modify -remove` command:

- The `symcg modify -remove` command cannot remove SRDF groups from a consistency group.
- The `symcg modify -remove` command prohibits a cascaded R1 device from being removed from a consistency group enabled at the composite group level.
- The `symcg modify -remove` command cannot remove both legs of a concurrent R11 device if they are enabled for consistency protection by the same SRDF group name.

Removing devices from an RDF1 consistency group

Table 24 shows the allowable device types for a dynamic modify remove operation on a composite group enabled for consistency protection at the composite group level and the SRDF group name level. This RDF1 CG is not concurrent or cascaded.

Table 24 Allowable device types for removing devices from an RDF1 CG

Device type in CG	Enabled at CG level	Enabled at SRDF group name level
Simple R1 (R1->R2)	Allowed	Allowed
Concurrent R11	Not applicable	Not applicable
Cascaded R1	Not applicable	Not applicable

Example

To remove devices `50` and `51` from `RDFG 100` of `R1CG` on Symmetrix 306 to `RDFG 101` in the staging area, type:

```
symcg -cg R1CG modify -remove -sid 306 -stg_rdfg 101 -devs 50:51 -cg_rdfg 100
```

Removing devices from a concurrent RDF1 consistency group

Table 25 shows the allowable device types for a dynamic modify remove operation on a concurrent R1 composite group enabled for consistency protection at the composite group level and the SRDF group name level.

Table 25 Allowable device types for removing devices from a concurrent RDF1 CG

Device type in CG	Enabled at CG level	Enabled at SRDF group name level
Simple R1 (R1->R2)	Allowed	Allowed
Concurrent R11	Not allowed	Only allowed if both mirrors are not enabled by the same SRDF group name.
Cascaded R1	Now allowed	Not allowed

Example

To remove devices `20` through `30` from SRDF groups `70` and `80` of `ConCG` on Symmetrix 306 into SRDF groups `71` and `81` in the staging area, type:

```
symcg -cg ConCG modify -remove -sid 306 -stg_rdfg 71,81 -devs 20:30 -cg_rdfg 70,80
```

Removing devices from a cascaded RDF1 consistency group

Table 26 shows the allowable device types for performing a dynamic modify remove operation on a cascaded R1 composite group enabled for consistency protection at the CG level and the SRDF group name level.

Table 26 Allowable device types for removing devices from a cascaded RDF1 CG

Device type in CG	Enabled at CG level			Enabled at SRDF group name level		
	Hop1 enabled Hop2 not enabled	Hop1 enabled Hop2 enabled	Hop1 not enabled Hop2 enabled	Hop1 enabled Hop2 not enabled	Hop1 enabled Hop2 enabled	Hop1 not enabled Hop2 enabled
Simple R1 (R1->R2)	Allowed	Not applicable	Not applicable	Allowed	Not applicable	Not applicable
Concurrent R11	Not allowed	Not allowed	Not allowed	Not allowed	Not allowed	Not allowed
Cascaded R1	Allowed	Allowed	Not allowed	Allowed	Allowed	Not allowed

Example

To remove device 20 of SRDF groups 38 (R1->R21) and 39 (R21->R2) of `CasCG` on Symmetrix 306 into SRDF groups 28 and 29 in the staging area, type:

```
symcg -cg CasCG modify -remove -sid 306 -cg_rdfg 38 -devs 20 -cg_r21_rdfg 39 -stg_rdfg 28
-stg_r21_rdfg 29
```

Recovering from a failed dynamic modify operation

After a dynamic modify operation is issued, Solutions Enabler stores all details about the command, such as the target CG, SRDF groups, staging area, and operation type in the Symmetrix File System (SFS). If a dynamic modify operation fails and all sites are reachable, reissue the command with the *exact* parameters. If the command fails again, execute the following command:

```
symcg modify -cg CasCG -recover
```

This command uses all existing dynamic modify command information in SFS. The recover operation either completes the unfinished steps of the dynamic modify operation or rolls back any tasks performed on the CG before failure, placing the CG into its original state. For example, if a concurrent R11 loses a link to one of its mirrors during a dynamic modify add operation, the recover operation may remove all devices added to the CG by this operation. This ensures that the CG device pairs are consistent at all three sites.

SRDF consistency with a parallel database

Figure 38 illustrates the use of an SRDF consistency group with a parallel database such as Oracle Parallel Server (OPS). The production database system spans two hosts and two Symmetrix arrays (Symmetrix A and C). A user-defined SRDF consistency group includes R1 devices from Symmetrix arrays A and C.

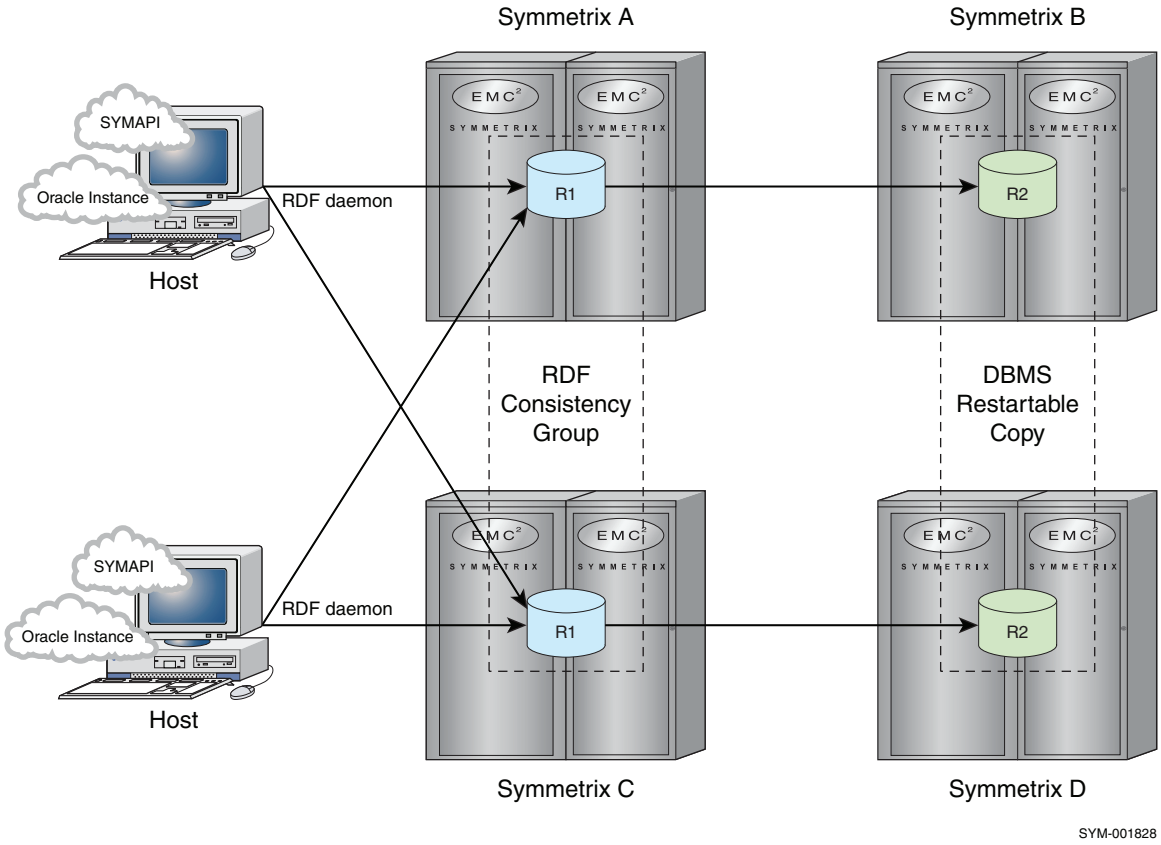


Figure 38 Using an SRDF consistency group with a parallel database configuration

The same consistency group definition must exist on both hosts. If enabled, Group Name Services (GNS) automatically propagates a composite group definition to the Symmetrix arrays and to all locally-attached hosts running the GNS daemon.

Although each production host can provide I/O to both R1 devices in the configuration, the DBMS has a distributed lock manager that ensures two hosts cannot write data to the same R1 device at the same time. The SRDF links to two remote Symmetrix arrays (Symmetrix B and D) enable the R2 devices on those systems to mirror the database activity on their respective R1 devices. A typical remote configuration includes a target-side host or hosts (not shown in the illustration) to restart and access the database copy at the target site.

Although Figure 38 shows the SRDF daemons located on the production hosts, it is recommended that the SRDF daemons be located on the control hosts that do not include the production application.

SRDF consistency with BCV access at the target site

When an SRDF consistency group includes devices on one or more source Symmetrix arrays propagating production data to one or more target Symmetrix arrays, TF BCVs at the target site can be indirectly involved in the consistency process.

Figure 39 illustrates a configuration with target-side BCVs that mirror the R2 devices. To access data on the BCVs from the target-side hosts, you need to split the BCV pairs at the target sites.

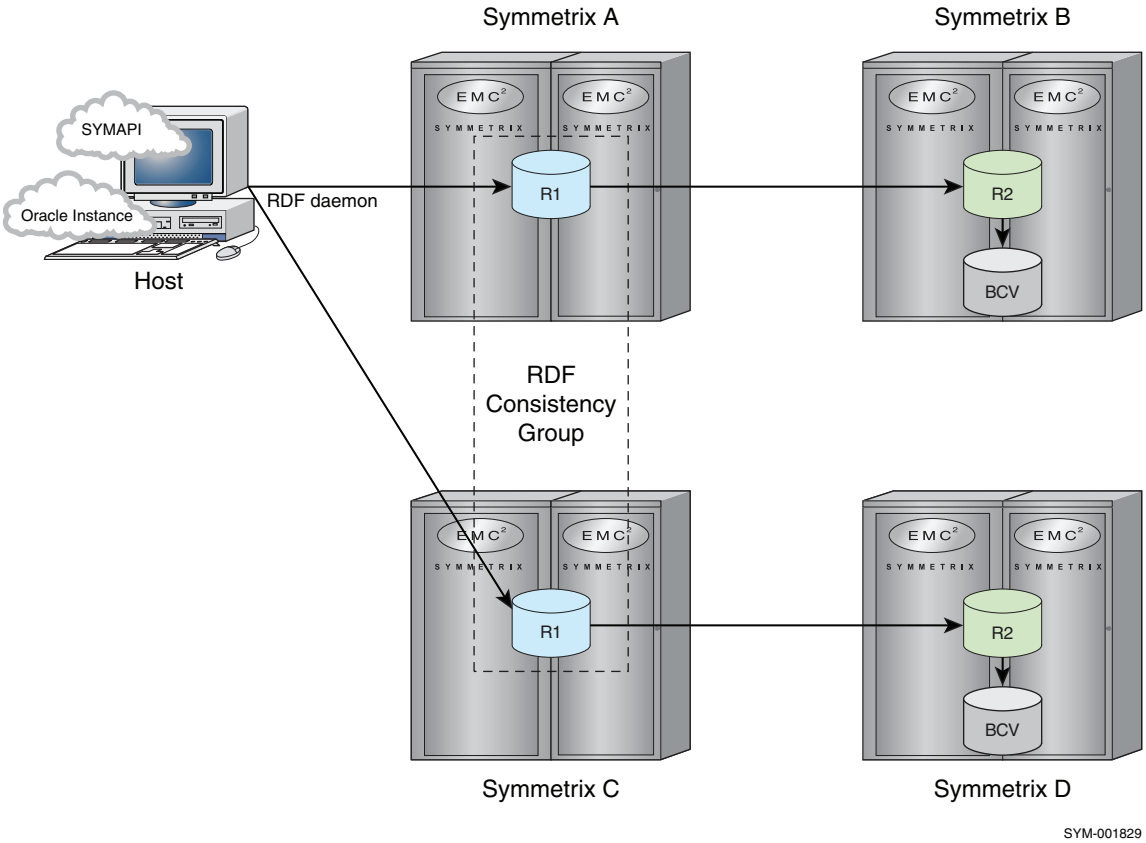


Figure 39 Using an SRDF consistency group with BCVs at the target site

The recovery sequence in a configuration that includes BCVs at the target site is the same as described in the previous section except that, at the end of the sequence, the DBMS-restartable copy of the database exists on the target R2 devices and on the BCVs if the BCVs were synchronized with the target site's R2 devices at the time the interruption occurred.

When data propagation is interrupted, the R2 devices of the suspended SRDF pairs are in a Write Disabled state. The target-side hosts cannot write to the R2 devices, thus protecting the consistent DBMS-restartable copy on the R2 devices. Splitting off the BCV version of the restartable copy allows you to perform disaster testing or business continuance tasks on that data while still maintaining an unchanged R2 copy of the database that can remain consistent with the R1 production database until normal SRDF mirroring between the R1 and R2 sides resumes.

This configuration provides a way to split off and access the DBMS-restartable database copy on the BCVs without risking the data protection that exists on the R2 devices when propagation of data is interrupted.

To manage the BCVs from the R2 side, you can associate the BCVs with a single SRDF consistency group defined on the target-site host that is connected to Symmetrix arrays B and D.

Although Figure 39 on page 226 shows the SRDF daemon located on the production host, it is recommended that the SRDF daemon be located on control hosts that do not include the production application.

CHAPTER 10
SRDF Device Migration

This chapter describes how to migrate data from an existing R1 or R2 device to a new device, and then replace the existing device with the new device in an SRDF pair. Topics include:

- Overview 230
- R1 and R2 migration procedures 236
- Applicable SRDF pair states for migration 247

Overview

The SRDF device migration feature provides the ability to replace an existing device with in an SRDF pair with a new device on a different Symmetrix array. During migration, a concurrent SRDF relationship is established to transfer data from an existing R1 device to a new device in adaptive copy disk mode. After this data transfer, the R1 device or the R2 device is replaced with the newly-populated device in the SRDF pair.

Configuration requirements

The following are the configuration requirements for an SRDF device migration:

- The three Symmetrix arrays must be running Enginuity 5670 or higher.
- The three Symmetrix arrays must be distinct with a unique Symmetrix ID (sid).
- The existing SRDF device and the new devices must be dynamic R1 or R2 capable (DRX devices).
- If replacing an R1 or an R2 device in an SRDF pair running in SRDF/A mode, the existing R1 or R2 Symmetrix array must be running Enginuity 5671 or higher. This is because Enginuity 5670 only supports one SRDF group that is SRDF/A capable.

R1 device migration

This section explains and illustrates the following processes involved in an R1 migration:

- Setting up the required configuration, including SRDF groups and devices.
- Establishing a concurrent SRDF relationship for data transfer to the new device.
- Replacing the R1 device with the newly-populated device in an SRDF pair.

For instructions on performing an R1 migration, refer to "R1 and R2 migration procedures" on page 236.

Configuration setup

Figure 40 on page 231 shows a sample configuration for an R1 migration. Symmetrix A contains the existing R1 device paired with the R2 device in Symmetrix B, and Symmetrix C contains the new non-SRDF device that will replace the existing R1 device. As indicated by the dotted lines, no SRDF relationship exists with Symmetrix C.

Temporary SRDF group for R1 migration (only)

A temporary SRDF group is required to synchronize data from the existing R1 device to the new device.

In Figure 40 on page 231, the SRDF group 17 (RDFG 17) is used to synchronize data from the existing R1 to the new device in Symmetrix C. This new device will eventually replace the existing R1 device during the migration replace process.

Note: In this sample configuration, the user is responsible for creating RDFG 72 before migration, and then removing it after the migration is finished.

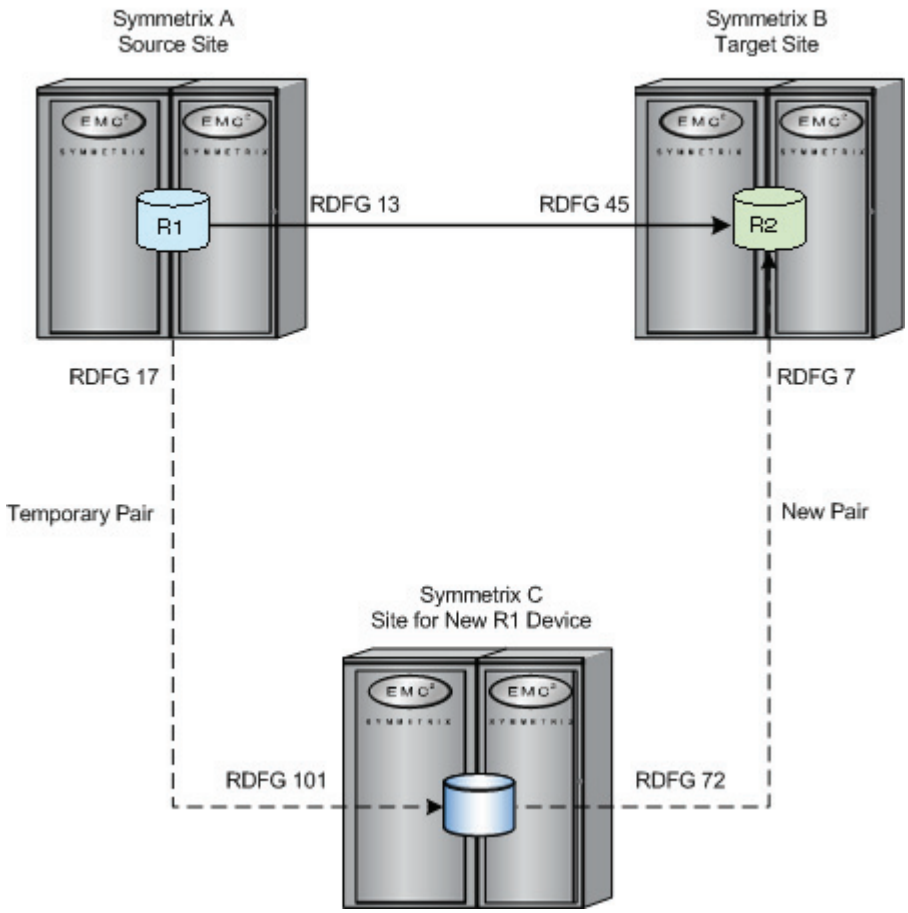


Figure 40 Configuration setup for an R1 migration

Establish a concurrent SRDF relationship

During this process, you are preparing to replace the R1 device in the existing SRDF pair by establishing a concurrent SRDF relationship to transfer data from the existing source device to two target devices. The `symrdf migrate -setup` command establishes this concurrent SRDF relationship among the three sites shown in Figure 41 on page 232. The R1 device becomes the concurrent R11 device writing to two target devices.

Data synchronization in adaptive copy disk mode begins between the device and the R2 device on Symmetrix C, which will become the source for Symmetrix B. As shown by the dotted line, no SRDF pairing exists between the devices on Symmetrix C and Symmetrix B.

After the two R2 devices are near synchronization with the R11 device, you must shut down all applications writing to the source device before replacing the R1 device. Additionally, any existing device group or composite group scripts my need to be modified to deal with the new configuration.

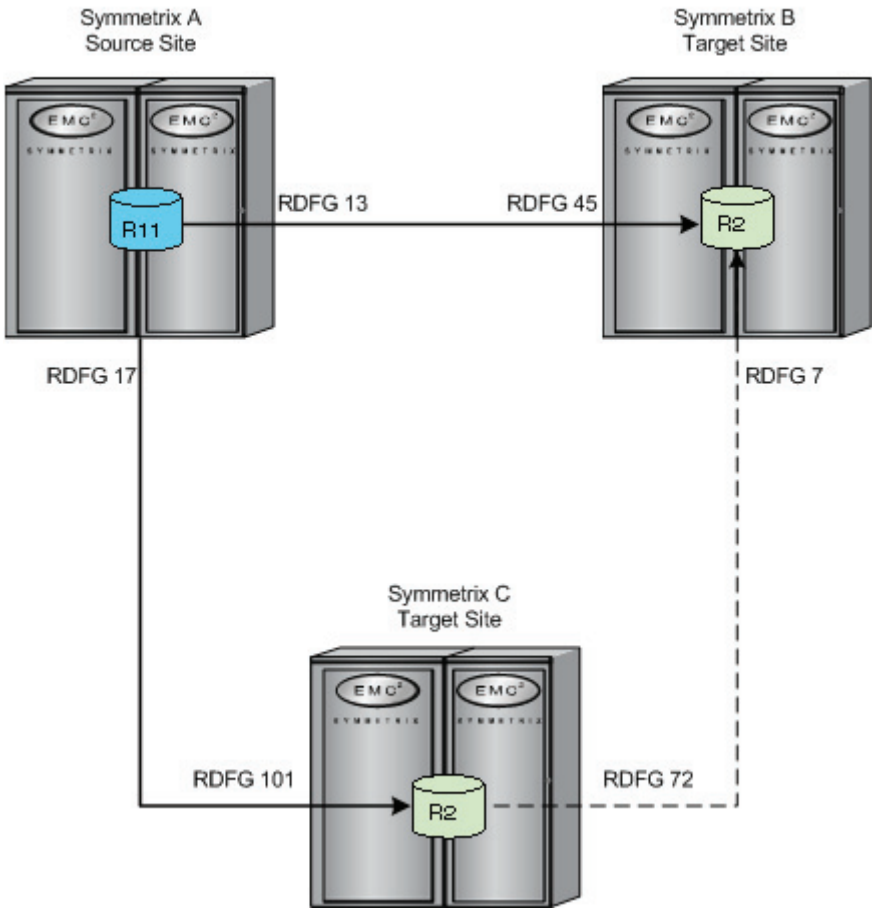


Figure 41 Establishing a concurrent SRDF relationship during an R1 migration

Replace R1

During the replace process, you are replacing the existing R1 (R11) with the new R1 device in the SRDF pair. Before replacing the source device, the R11 device and the R2 device must be close to synchronization, and all applications writing to the source site must be shut down.

The SYMCLI performs the following actions when you issue the `symrdf migrate -replace R1` command:

1. Sets the source device to USR-NR (user not ready), which prevents applications writing to or reading from the R1 device.
2. Verifies the devices are in the correct pair state for replacement. For more information, refer to "Applicable SRDF pair states for migration" on page 247.
3. Waits until all invalid tracks are cleared and drains the SRDF/A session (if applicable). When this is achieved, step 4 executes.

4. Removes the SRDF pairing between the devices on Symmetrix A and B.
5. Removes the SRDF pairing between the devices on Symmetrix A and C.
6. Sets an SRDF pairing between the devices on Symmetrix C and B using the original SRDF mode of Symmetrix A and B. No additional copying of data is required between this SRDF pair because this command ensures data is the same on both devices. The devices are read/write on the SRDF links and the R1 device is ready.

At this point, you can restart the applications writing to the new R1 device on Symmetrix C. This begins the processing of I/Os from the host and starts data flow from Symmetrix C to Symmetrix B. The original R1 devices remain USR-NR.

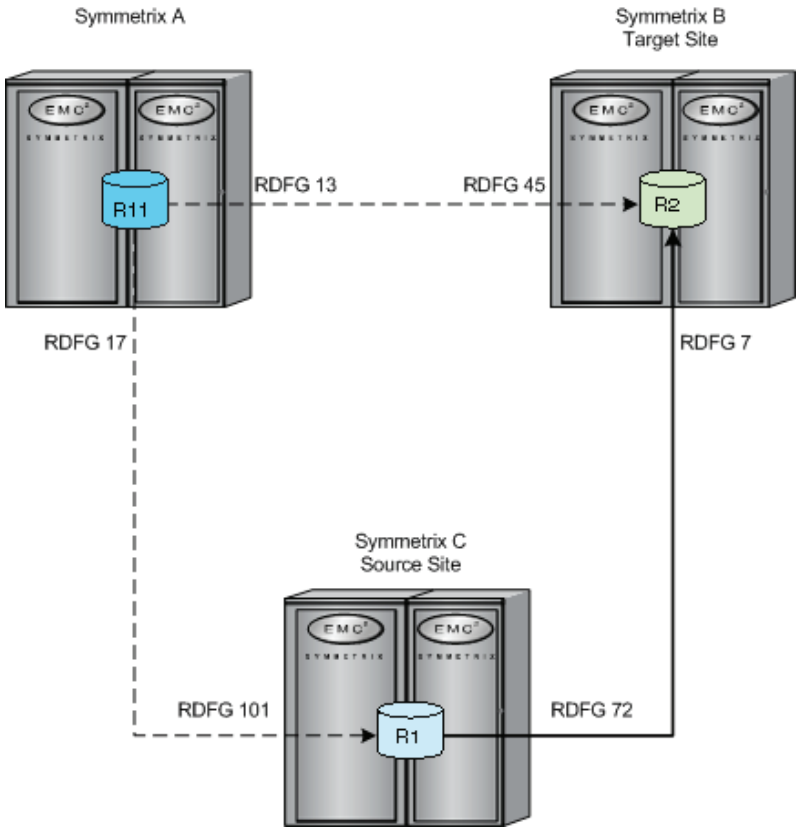


Figure 42 Replacing the source device during an R1 migration

R2 device migration

This section explains and illustrates the following processes involved in an R2 migration:

- Setting up the required configuration, including SRDF groups and devices.
- Establishing a concurrent SRDF relationship for data transfer to the new device.
- Replacing the R2 device with a newly-populated device in an SRDF pair.

For instructions on performing an R1 migration, refer to "R1 and R2 migration procedures" on page 236.

Configuration setup

Figure 43 shows a sample configuration for an R2 migration. Symmetrix A contains the R1 device paired with the existing R2 device in Symmetrix B, and Symmetrix C contains the new non-SRDF device that will replace the R2 device. As indicated by the dotted lines, no SRDF pairing exists with Symmetrix C.

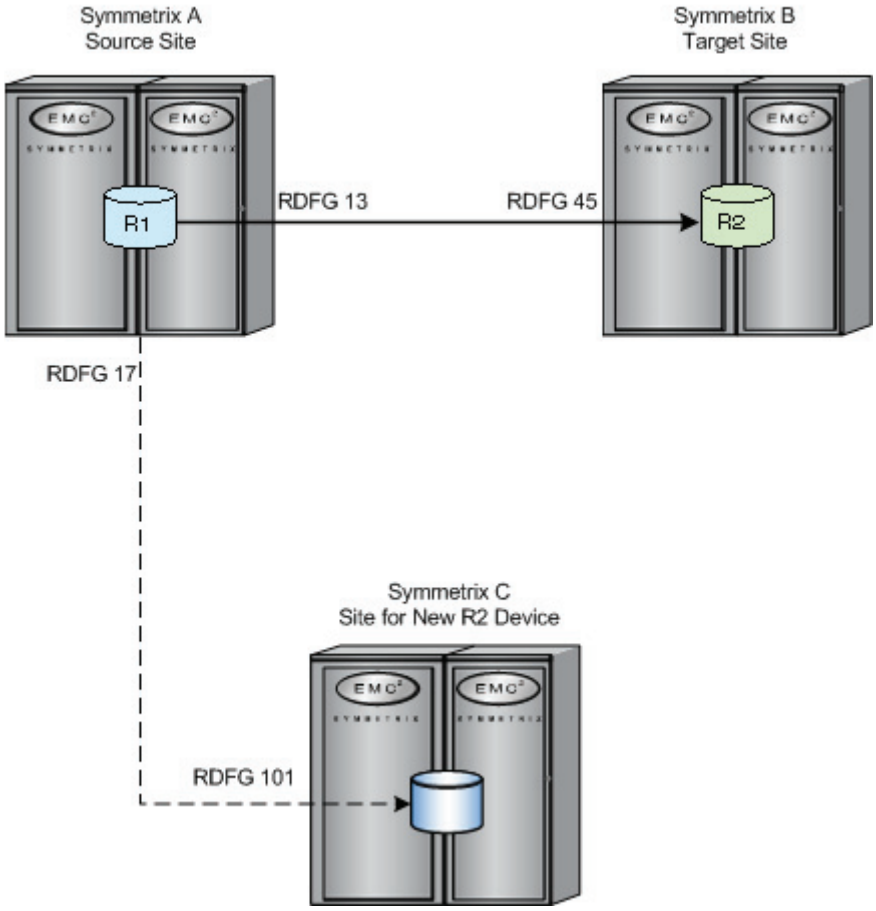


Figure 43 Configuration setup for an R2 migration

Establish a concurrent SRDF relationship

During this process, you are preparing to replace the R2 device in an existing SRDF pair by establishing a concurrent SRDF relationship to transfer data from the existing source device to two target devices.

Because you are replacing an R2 device in an SRDF pair (not an R1 device), the source site continues to accept I/Os from the host, and there is no need to shut down the applications writing to R1. Also, an R2 migration does not require a temporary pairing like an R1 migration, and the source and target devices do not have to be close to synchronization.

The `symrdf migrate -setup` command establishes this concurrent SRDF relationship among the three sites shown in Figure 44. The R1 becomes the R11 device writing to two target R2 devices. Additionally, any existing device group or composite group scripts may need to be modified to deal with R11.

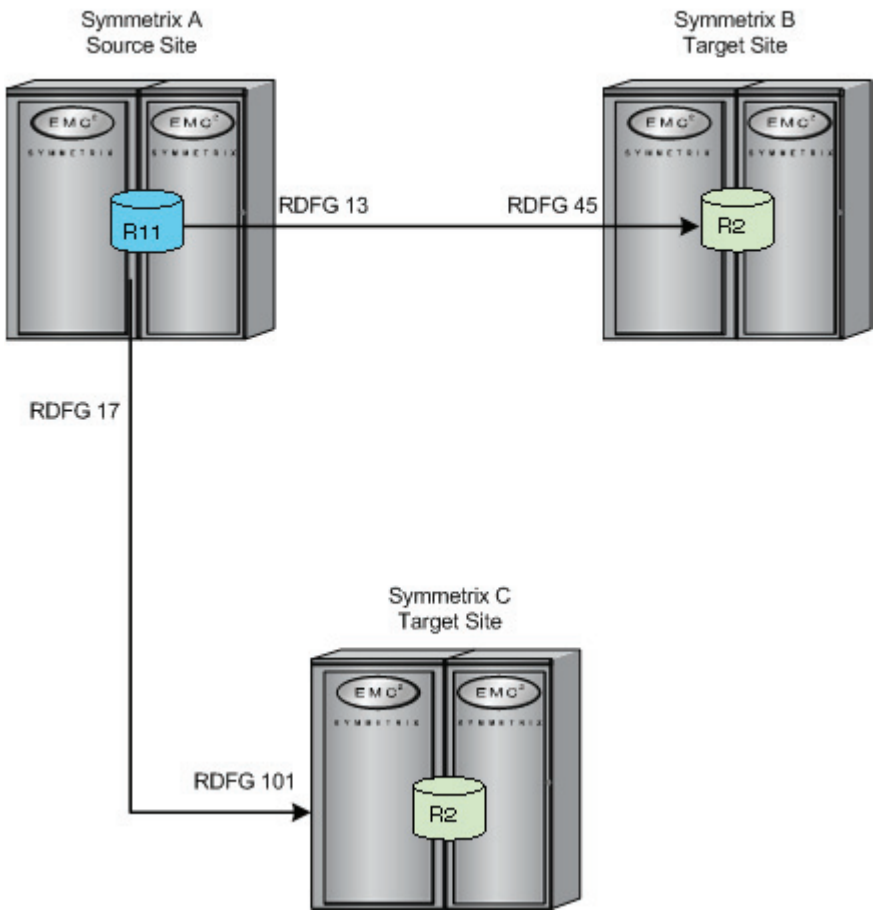


Figure 44 Establishing a concurrent SRDF relationship during an R2 migration

Replace R2

During the replace process, you are replacing the existing R2 device with the new R2 device in the SRDF pair shown in Figure 45.

The SYMCLI performs the following actions when you issue the `symrdf migrate -replace R2` command:

1. Verifies the devices are in the correct pair state for replacement. For more information, refer to "Applicable SRDF pair states for migration" on page 247.
2. Removes the SRDF pairing between the devices on Symmetrix A and B.

3. Sets the mode of Symmetrix A and C using the original SRDF mode of Symmetrix A and B.

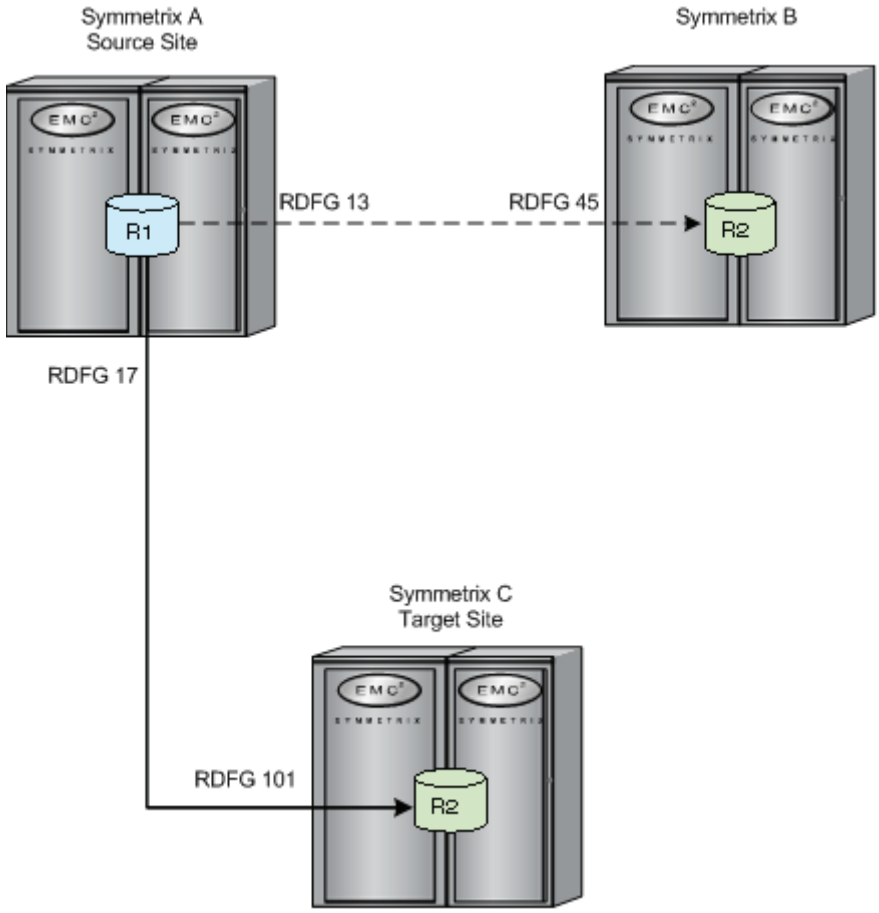


Figure 45 Replacing a target device during an R2 migration

R1 and R2 migration procedures

This section explains how to perform an R1 and an R2 migration using sample SRDF configurations.

Before you begin

Complete the following tasks before beginning the migration procedure:

- Plan accordingly for each migration. If you have defined scripts for your existing R1/R2 pair, evaluate how you may need to modify those scripts with new Symmetrix IDs, SRDF device pairings, device groups, composite groups, and so on. Keep in mind that during a device migration, the R1/R2 pair transforms into a concurrent SRDF relationship (R2<-R11->R2), and then back into an R1->R2 relationship.
- An SRDF group must exist for the new device. If R1 is being replaced, this is the SRDF group between the new R1 and the existing R2. If the R2 is being replaced, this is the SRDF group between the new R2 and the existing R1.

- For an R1 migration *only*, a temporary SRDF group is required to synchronize data from the existing R1 device to the new device. If performing an R1 migration, create this temporary SRDF group.
- Before replacing the R1 device, you must shut down all applications using it. When replacing an R2 device, its related R1 device can remain online with its host.
- Review "Applicable SRDF pair states for migration" on page 247.

Restrictions and limitations

The following are the restrictions and limitations for migrating to a new device.

SRDF/A device pairs

- The attributes associated with an existing SRDF group pertaining to an SRDF/A session are not automatically associated with the new SRDF group after migration. You must issue the `symconfigure` command on the new SRDF group and set the appropriate attributes, such as the minimum_cycle_time and the DSE (Delta Set Extension) autostart settings.
- If replacing a device of an SRDF pair in SRDF/A mode, all existing rules for DSE apply if DSE autostart is enabled on the new SRDF group. An example of such a rule is the DSE threshold must be less than the maximum cache usage for the new SRDF group.
- If replacing the R1 device of an SRDF pair in SRDF/A mode, the new SRDF group in the new R1 Symmetrix array must be SRDF/A capable.
- If replacing a device of an SRDF pair in SRDF/A mode and Cache partitioning is enabled on the new Symmetrix array, all new devices must belong to the same cache partition.
- If the existing device is in SRDF/A mode, the entire SRDF group must be migrated.
- If the existing device is in SRDF/A mode, the new SRDF group must be empty. If replacing the R1 device, the temporary SRDF group must not be in SRDF/A mode.
- The existing SRDF device pair cannot be in semi-synchronous mode.

Devices

- The new device (R1 or R2) cannot be an SRDF device before migration.
- The existing device (R1 or R2) and the replacement device cannot be diskless.
- The new R1 device cannot be larger than the existing R1 device.
- The existing R1 device cannot have any local invalid tracks.
- After migration, the R2 device cannot be larger than the R1 device.
- The existing (R1 or R2) and the new device cannot be configured for SRDF/Star.
- The existing device and the replacement device cannot be a source or a target device for TF/Mirror, TF/Snap, TF/Clone, Open Replicator, and Federated Live Migration. This restriction does not apply to the SRDF partner of the existing device.

- The existing R1/R2 device pair cannot be in a concurrent or cascaded SRDF relationship. To indicate this pair is not part of such a configuration, set the `-config` option to equal `pair` in `symrdf migrate -setup`.
- An SRDF consistency protection group must be enabled at the RDFG-name level, **NOT** at the composite-group level. Otherwise, when you issue the `migrate -setup` command, SYMCLI stops the monitoring/cycle switching of your composite group. The next section, "Sample R1 migration procedure", explains what to do if you have an SRDF consistency protection group enabled at the composite-group level.

Sample R1 migration procedure

This section describes the steps for migrating R1 devices to new devices using a sample configuration. For this sample procedure, the SRDF consistency protection group is enabled at the composite-group level. This provides you with the steps to follow to change this setting and enable SRDF consistency protection at the RDFG-name level.

Figure 46 shows an R1 and R2 relationship between Symmetrix 43 and Symmetrix 90. After R1 migration, the devices in Symmetrix 306 will become the source devices for Symmetrix 90.

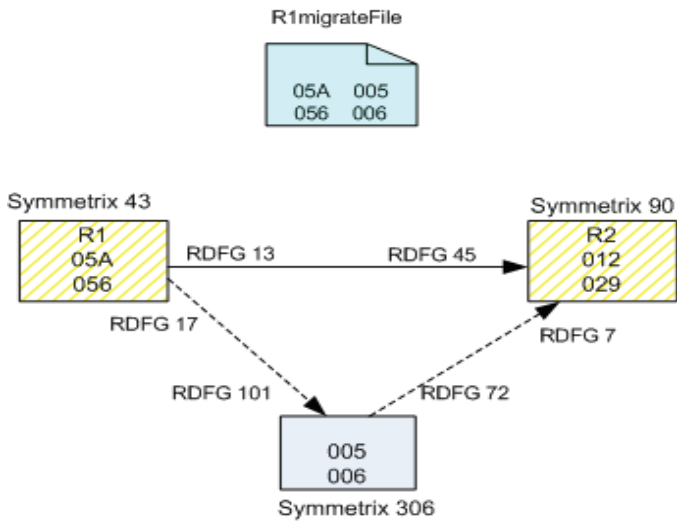


Figure 46 R1 migration example: Initial configuration

Step 1: Querying the sample SRDF/A configuration

This step shows you how to query a configuration with SRDF consistency protection enabled at the composite-group level. To maintain consistency protection after establishing a concurrent SRDF relationship, you must remove this setting, and then enable consistency protection at the RDFG-name level, explained in "Step 2: Changing the SRDF consistency protection setting" on page 239.

```
symrdf -cg MigrateRDF query -detail

Composite Group Name       :  MigrateRDF
Composite Group Type       :  RDF1
Number of Symmetrix Units  :  1
Number of RDF (RA) Groups  :  1
RDF Consistency Mode       :  MSC
```

```
RDFA MSC Consistency Info:{
    Session Status                      : Active
    Consistency State                   : CONSISTENT
    }

Symmetrix ID                      : 000192600043 (Microcode Version: 5874)
Remote Symmetrix ID               : 000192600090 (Microcode Version: 5874)
RDF (RA) Group Number             :   1 (00) 13 (0C)
RDFA Info:
    {
    Cycle Number                        : 29
    Session Status                      : Active - MSC
    Consistency Exempt Devices          : No
    Minimum Cycle Time                  : 00:00:30
    Avg Cycle Time                      : 00:00:30
    Duration of Last cycle              : 00:00:30
    Session Priority                    : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1           : 00:00:42
    R2 Image Capture Time               : Mon Sep 21 13:28:44 2009
    R2 Data is Consistent               : True
    R1 Side Percent Cache In Use        : 0
    R2 Side Percent Cache In Use        : 0
    R1 Side DSE Used Tracks             : 0
    R2 Side DSE Used Tracks             : 0
    Transmit Idle Time                  : 00:00:00
    }

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   005A NR       0       0 RW 0012 WD       0       0 A..X. Consistent
DEV002   00F8 NR       0       0 RW 0029 WD       0       0 A..X. Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0
```

Step 2: Changing the SRDF consistency protection setting

For this sample configuration, you are setting the SRDF group name, `siteb`, disabling SRDF consistency protection at the composite-group level, and then enabling it at the RDFG-name level.

```
symcg -cg MigrateRDF -rdfg 043:13 set -name siteb
symcg -cg MigrateRDF disable

A consistency 'Disable' operation execution is
in progress for composite group 'MigrateRDF'. Please wait...

The consistency 'Disable' operation successfully executed for
composite group 'MigrateRDF'.

symcg -cg MigrateRDF -rdfg name:siteb enable

A consistency 'Enable' operation execution is
in progress for composite group 'MigrateRDF'. Please wait...

The consistency 'Enable' operation successfully executed for
composite group 'MigrateRDF'.
```

Verifying the changes

The following command verifies that the changes and additions were made to the SRDF/A configuration. As shown in this partial listing, SRDF consistency protection is now enabled using the SRDF group name of `siteb`.

```
symrdf -cg MigrateRDF query -detail

Composite Group Name      :  MigrateRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE

RDFG Names:
    {
    RDFG Name                          : siteb
    RDF Consistency Mode               : MSC
    MSC Consistency Info:
        {
        Session Status                 : Active
        Consistency State              : Consistent
        }
    }
```

Step 3: Pairing devices

For an R1 migration, you create a device file to pair SRDF devices with the new non-SRDF devices. This pairing is used temporarily to transfer data from the existing R1 devices to the devices that will eventually replace them in an SRDF pair.

For example, the first column of the R1migrateFile device file contains the existing R1 devices paired with the new non-SRDF devices in the second column. The R1 devices 05A and 056 in Symmetrix 43 are paired with the new devices 005 and 006 in Symmetrix 306. The pairing of these devices becomes obsolete after the Symmetrix 43 devices are replaced by the Symmetrix 306 devices.

R1migrateFile	
#R1 devices	New devices
05A	005
056	006

Step 4: Establishing a concurrent SRDF relationship

The `symrdf migrate -setup` command establishes a concurrent SRDF relationship between the existing R1 devices and the new devices in adaptive copy disk mode, and begins the synchronization of these devices.

Note: At this point, any existing device group or composite group scripts may need to be modified to deal with (temporary) change of the existing R1 devices to R11 devices.

The following command establishes a concurrent SRDF relationship between the R1 devices in Symmetrix 43 and the new devices in Symmetrix 306 using SRDF group 17. This is a temporary relationship to transfer data from the existing R1 to its replacement.

Using the -force option

For this example, you must use the `-force` option because SRDF consistency protection is enabled. You will disable SRDF consistency protection in order to replace R1, as shown in"Step 5: Replacing R1 devices with new devices" on page 242.

```
symrdf -sid 043 -rdfg 17 -f R1MigrateFile migrate -setup -config pair

An RDF 'Migrate Setup' operation execution is
in progress for device file 'R1migrateFile'. Please wait...

Cannot proceed in the current RDF Consistency state except if the force flag is used

symrdf -sid 043 -rdfg 17 -f R1MigrateFile migrate -setup -config pair -force

An RDF 'Migrate Setup' operation execution is
in progress for device file 'R1migrateFile'. Please wait...

    Migrate Setup for R1 device(s) in (043,017).......................Started.
    Create RDF Pair in (0043,017)....................................Started.
    Create RDF Pair in (0043,017)....................................Done.
    Mark target device(s) in (0043,017) for full copy from source....Started.
    Devices: 06F0-06FF in (0043,017)............................... Marked.
    Mark target device(s) in (0043, 017) for full copy from source...Done.
    Merge track tables between source and target in (0043,017).......Started.
    Devices: 06F0-06FF in (0043,017)............................... Merged.
    Merge track tables between source and target in (0043,017).. ....Done.
    Resume RDF link(s) for device(s) in (0043,017)...................Started.
    Resume RDF link(s) for device(s) in (0043,017)...................Done.
    Migrate Setup for R1 device(s) in (0043,017) ....................Done.

The RDF 'Migrate Setup' operation finished successfully
for device file 'R1MigrateFile'.
```

Note: If the host is reading and writing to the R1 device during this action, a synchronized pair state may not be attainable because the pair is operating in adaptive copy disk mode.

As shown in Figure 47, devices 05A and 056 are paired with devices 005 and 006 in a concurrent SRDF relationship using SRDF group 17. Devices 005 and 006 are made read/write on the SRDF links in adaptive copy disk mode. Because this is an R1 device migration, the SRDF group 17 is used temporarily to transfer data from the R1 devices to the new devices.

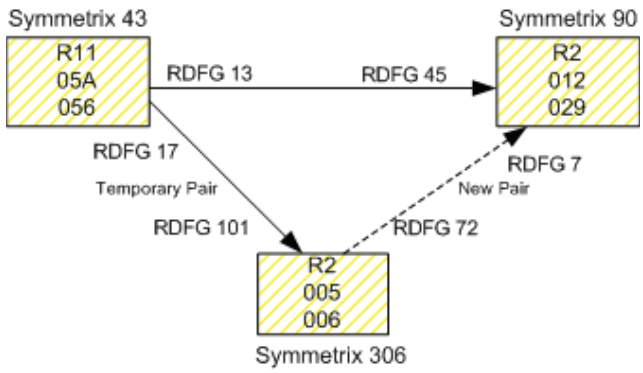


Figure 47 Concurrent SRDF relationship

Step 5: Replacing R1 devices with new devices

Before replacing R1 devices, you must disable consistency (if enabled), as shown below, and terminate any TF/Mirror, TF/Snap, TF/Clone, Open Replicator, and Federated Live Migration sessions. To disable SRDF consistency protection for composite group `MigrateRDF`, type:

```
symcg -cg MigrateRDF -rdfg name:siteb disable

A consistency 'Disable' operation execution is
in progress for composite group 'MigrateRDF'. Please wait...

The consistency 'Disable' operation successfully executed for
composite group 'MigrateRDF'.
```

The `symrdf migrate -replace` command sets the R1 (R11) device as USR-NR, completes the final synchronization of data between the existing and the new device (if needed), and reconfigures the devices into a new SRDF pair. The device pairings of the replaced devices are removed. The new devices become R1 devices paired with the existing R2 devices using the original SRDF mode of the replaced pair.

Note: The `migrate -replace R1` command waits for synchronization to finish and can run for a considerable amount of time. To avoid the locking of the SYMAPI database for this entire time, set the environment variable SYMCLI_CTL_ACCESS=PARALLEL. If you set this variable, you may need to run the `symcfg sync` command after the R1 migration is complete.

The following command uses the new SRDF group 72 to reconfigure and connect the new R1 devices 005 and 006 in Symmetrix 306 with the R2 devices 012 and 029 in Symmetix 90.

```
symrdf -sid 043 -rdfg 17 -f R1migrateFile migrate -replace r1 -config pair -new_rdfg 72

An RDF 'Migrate Replace R1' operation execution is
in progress for device file 'R1migrateFile'. Please wait...

    Migrate Replace R1 for new R1 device(s) in (0306, 072)...........Started.
    Waiting for invalid tracks to reach 0 in   (0043, 013)...........Started.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........Started.
    Waiting for invalid tracks to reach 0 in   (0043, 013)...........Done.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........915994 remaining.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........519572 remaining.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........245889 remaining.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........107613 remaining.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........1110 remaining.
    Waiting for invalid tracks to reach 0 in   (0043, 017)...........Done.
    Suspend RDF link(s) for device(s) in (0043,013)..................Started.
    Suspend RDF link(s) for device(s) in (0041,013)..................Done.
    Suspend RDF link(s) for device(s) in (0043,017)..................Done.
    Delete RDF Pair in (0043,013)....................................Started.
    Delete RDF Pair in (0043,017)....................................Started.
    Delete RDF Pair in (0043,013)....................................Done.
    Delete RDF Pair in (0043,017)....................................Done.
    Create RDF Pair in (0306,072)....................................Started.
    Create RDF Pair in (0306,072)....................................Done.
    Resume RDF link(s) for device(s) in (0306,072)...................Started.
    Merge track tables between source and target in (0306,072).......Started.
    Devices: 0690-069F in (0306,072)............................... Merged.
    Merge track tables between source and target in (0306,072).......Done.
    Resume RDF link(s) for device(s) in (0306,072)...................Done.
    Migrate Replace R1 for new R1 device(s) in (0306, 072)...........Done.
```

```
The RDF 'Migrate Replace R1' operation finished successfully
for device file 'R1migrateFile'.
```

Note: After replacing the R1 devices, you need to recreate your device groups and/or composite groups, possibly update your scripts since the devices are no longer concurrent SRDF. Any TF/Mirror, TF/Snap, TF/Clone, Open Replicator, and Federated Live Migration sessions used on the original R1 devices must be recreated on the new R1 devices.

For this example, you must delete the `MigrateRDF` consistency group and rebuild it, as shown in the following control operations:

```
symcg -force delete MigrateRDF
symcg create MigrateRDF -type rdf1 -rdf_consistency
symcg -cg MigrateRDF -sid 306 -rdfg 72 addall dev
symcg -cg MigratRDF enable

A consistency 'Enable' operation execution is
in progress for composite group 'MigrateRDF'. Please wait...

The consistency 'Enable' operation successfully executed for
composite group 'MigrateRDF'.
```

When migration is complete, the Symmetrix 306 devices become the R1 devices and are paired with the R2 devices in Symmetrix 90, as shown in Figure 48. This new SRDF pair uses the original SRDF mode of the replaced pair.

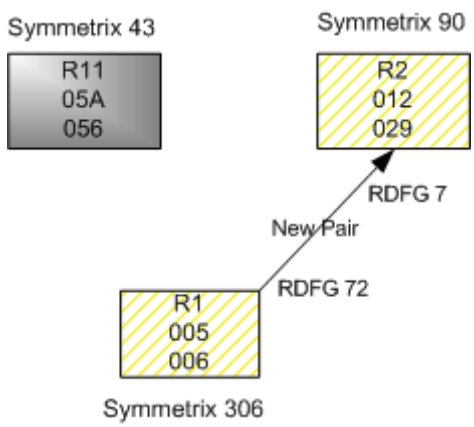


Figure 48 Migrated R1 devices

Step 6: Verifying the new pair and setting changes

The following command enables you to validate that the Symmetrix 306 devices are now the source devices for Symmetrix 90, and consistency protection is rebuilt.

```
symrdf -cg MigrateRDF query -detail

Composite Group Name       :  MigrateRDF
Composite Group Type       :  RDF1
Number of Symmetrix Units  :  1
Number of RDF (RA) Groups  :  1
RDF Consistency Mode       :  MSC

RDFG MSC Consistency Info:{
   Session Status                  : Active
```

```
   Consistency State              : CONSISTENT
    }

Symmetrix ID                      : 000190100306 (Microcode Version: 5773)
Remote Symmetrix ID               : 000192600090 (Microcode Version: 5874)
RDF (RA) Group Number             :   3 (02) - siteb
RDFA Info:
    {
    Cycle Number                      : 3
    Session Status                    : Active - MSC
    Consistency Exempt Devices        : No
    Minimum Cycle Time                : 00:00:30
    Avg Cycle Time                    : 00:00:33
    Duration of Last cycle            : 00:00:30
    Session Priority                  : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1         : 00:00:34
    R2 Image Capture Time             : Mon Sep 21 13:52:03 2009
    R2 Data is Consistent             : True
    R1 Side Percent Cache In Use      : 0
    R2 Side Percent Cache In Use      : 0
    R1 Side DSE Used Tracks           : 0
    R2 Side DSE Used Tracks           : 0
    Transmit Idle Time                : 00:00:00
    }

        Source (R1) View            Target (R2) View       MODES
-------------------------------  ------------------------ ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   0005 RW       0       0 RW 0012 WD       0       0 A..X. Consistent
DEV002   0006 RW       0       0 RW 0029 WD       0       0 A..X. Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0
```

Sample R2 migration procedure

This section describes the steps to migrate R2 devices to new devices using a sample configuration.

Figure 49 shows an R1 and R2 relationship between Symmetrix 43 and Symmetrix 90. In this migration example, the devices in Symmetrix 306 will become the R2 devices for Symmetrix 43.

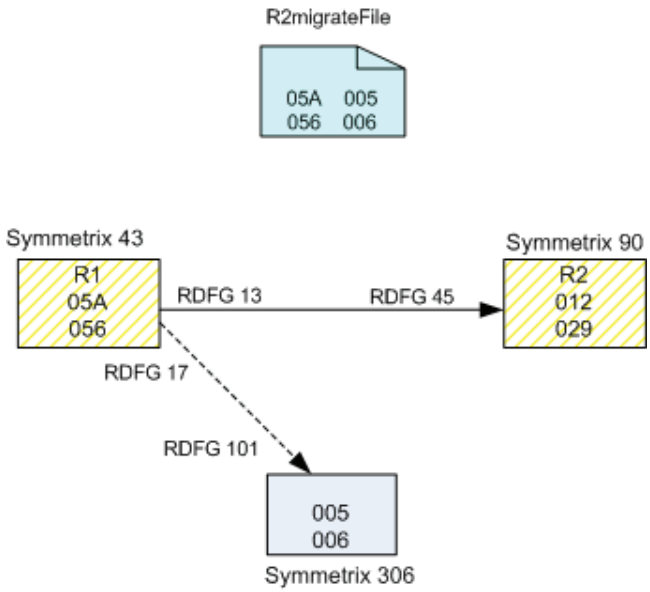


Figure 49 R2 migration example: Initial configuration

Step 1: Pairing devices

For an R2 migration, you create a device file to pair SRDF devices with the new non-SRDF devices, which will eventually replace the existing R2 devices.

For example, the first column of the following R2migrateFile device file contains the R1 devices that are paired with the R2 devices in Symmetrix 90 shown in Figure 49 on page 245. When migration is complete, these R1 devices will be paired with the new devices (Symmetrix 306) in the second column.

R2migrateFile	
#R1 devices	New devices
05A	005
056	006

Step 2: Establishing a concurrent SRDF relationship

The `symrdf migrate -setup` command establishes a concurrent SRDF relationship between the existing R1 devices and the new devices in adaptive copy disk mode, and begins the synchronization of these devices. Because this is an R2 migration, the R1 continues to process I/Os from its host, and synchronization is not required between the R1 and the new device.

Note: At this point, any existing device group or composite group scripts may need to be modified to deal with the (temporary) change of the existing R1 devices to R11 devices.

The following command establishes a concurrent SRDF relationship between the R1 devices 05A and 056 in Symmetrix 43 and the new devices 005 and 006 in Symmetrix 306 using SRDF group 17:

```
symrdf -file R2migrateFile -sid 043 -rdfg 17 migrate -setup -config pair
```

As shown in Figure 50, during migration setup, devices 05A and 056 are paired with devices 005 and 006 in a concurrent SRDF relationship using the SRDF group 17, Devices 005 and 006 are made read/write on the SRDF links in adaptive copy disk mode. Unlike an R1 device migration, the SRDF group 17 is used permanently to synchronize data from the source to the target devices.

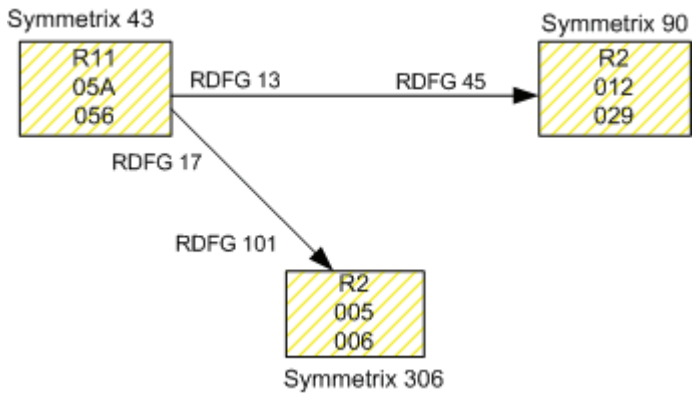


Figure 50 Concurrent SRDF relationship

Step 3: Replacing R2 devices with new devices

Note: Before replacing R2, you must disable SRDF consistency protection (if enabled), and terminate any TF/Mirror, TF/Snap, TF/Clone, Open Replicator, and Federated Live Migration sessions.

The `symrdf migrate -replace R2` command deletes the SRDF pairing between Symmetrix 43 and Symmetrix 90.

Note: After replacing R2, you must modify device groups and/or composite groups to remove all BCVs, VDEVS, TGTs from the original R2 and then add appropriate counterparts to the new R2. You must also recreate any TF/Mirror, TF/Snap, TF/Clone, Open Replicator, and Federated Live Migration sessions on the new R2.

The following command uses the SRDF group 17 to reconfigure and connect the R1 devices 05A and 056 with the new R2 devices 005 and 006:

```
symrdf -file R2migrateFile -sid 043 -rdfg 17 migrate -replace R2 -config pair
```

When migration is complete, the Symmetrix 306 devices become the R2 devices and are paired with the R1 devices in Symmetix 43 shown in Figure 51. This new pair uses the original SRDF mode of the replaced pair.

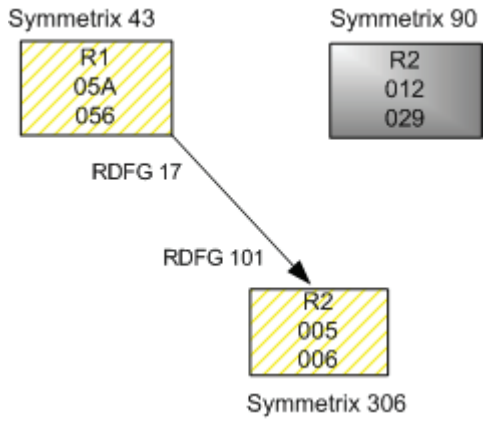


Figure 51 Migrated R2 devices

Applicable SRDF pair states for migration

An existing R1 and R2 pair must in a specific SRDF state to perform certain migration control operations.

Table 27 shows the applicable pair states for `symrdf migrate -setup` for an R1 and an R2 migration.

Table 27 SRDF migrate -setup control operation and applicable pair states

Control operation:	Pair state: existing R1->R2											
	SYNC IN PROG	SYNCHRONIZED	SPLIT	SUSPENDED	FAILED OVER	PARTITIONED[1]	PARTITIONED[2]	R1 UPDATED	R1 UPD IN PROG	INVALID	CONSISTENT	TRANSMIT IDLE
migrate -setup	P	P	P[3]	P[3]							P	

1. The remote Symmetrix array is in the SYMAPI database (it was discovered).
2. The remote Symmetrix array is not in the SYMAPI database (it was not discovered or was removed).
3. Only when replacing the R2 devices.

Pair states for migrate -setup

Figure 52 shows a sample configuration for an R1 migration. The R1 in Symmetrix A and the R2 in Symmetrix B must be in one of the applicable pair states before issuing the `symrdf migrate -setup` command, which establishes a concurrent SRDF relationship among the three sites.

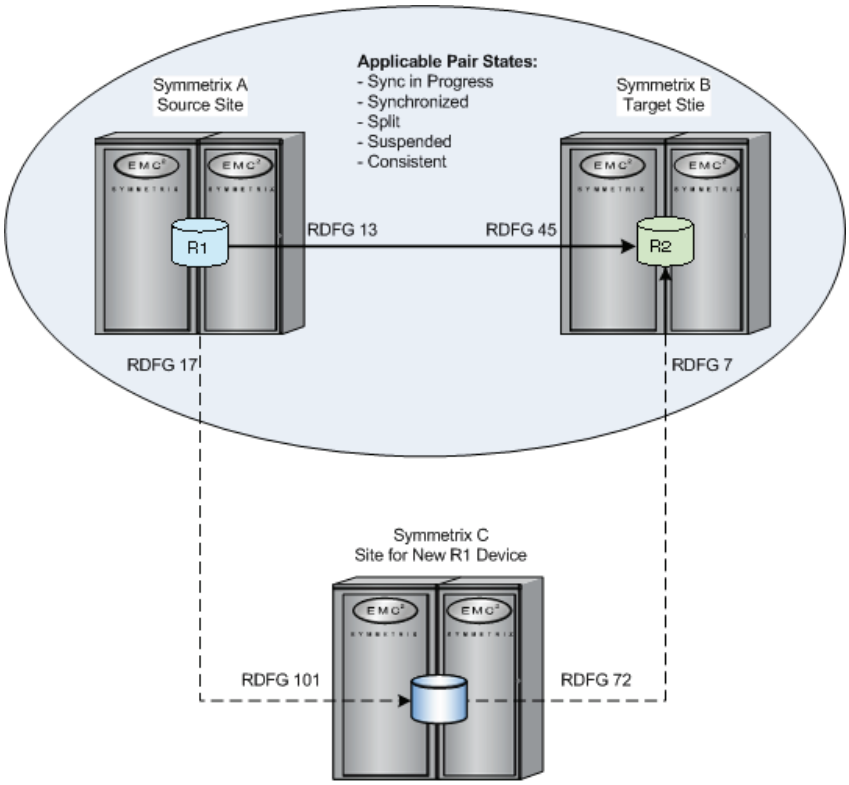


Figure 52 R1 migration: applicable R1/R2 pair states for migrate -setup

Figure 53 shows a sample configuration for an R2 migration. The R1 in Symmetrix A and the R2 in Symmetrix B must be in one of the applicable pair states before issuing the `symrdf migrate -setup` command, which establishes a concurrent SRDF relationship among the three sites.

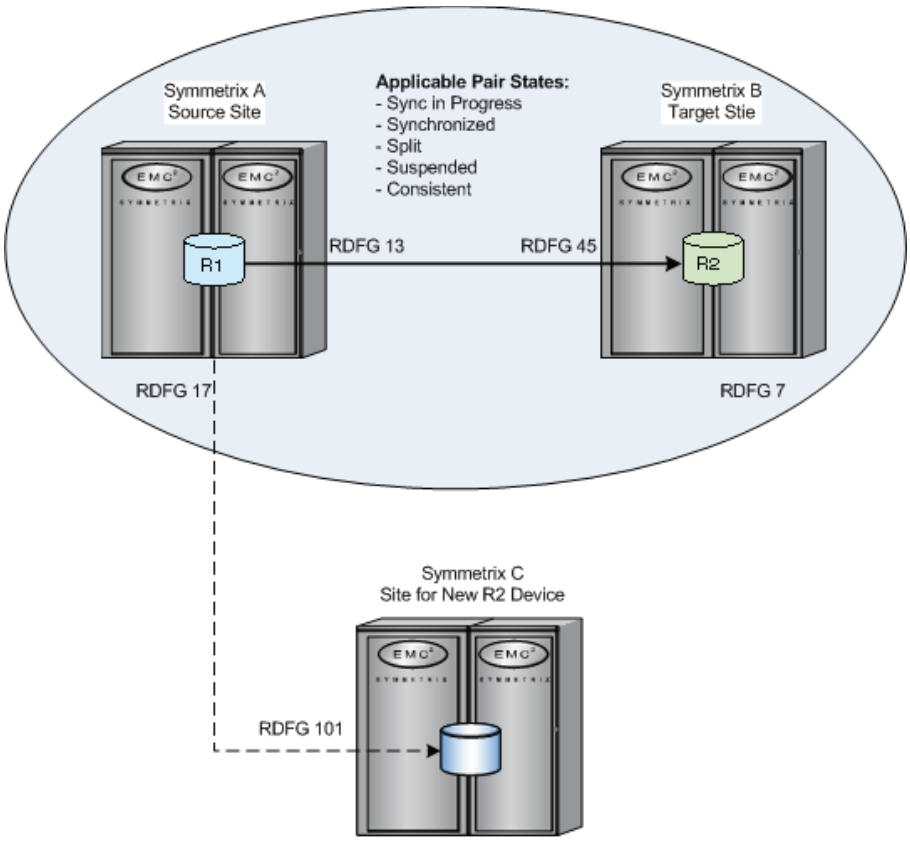


Figure 53 R2 migration: applicable R1/R2 pair states for migrate -setup

Pair states for migrate -replace for first leg of concurrent SRDF

Before replacing an R1, the R11 and its existing device must in a specific SRDF pair state shown in Figure 54. When replacing R2, the R11 and its existing R2 device must also be in a certain pair state shown in Figure 55. For the purpose of this discussion, this is the first leg of the concurrent SRDF relationship for both R1 and R2 migrations.

Table 28 shows the applicable pair states for `symrdf migrate -replace` for an R1 and an R2 migration.

Table 28 SRDF migrate -replace control operation and applicable pair states

Control operation:	Pair state: Existing ->R2											
	SYNC IN PROG	SYNCHRONIZED	SPLIT	SUSPENDED	FAILED OVER	PARTITIONED[1]	PARTITIONED[2]	R1 UPDATED	R1 UPD IN PROG	INVALID	CONSISTENT	TRANSMIT IDLE
migrate -replace	P	P	P	P							P	

1. The remote Symmetrix array is in the SYMAPI database (it was discovered).
2. The remote Symmetrix array is not in the SYMAPI database (it was not discovered or was removed).

Figure 54 shows a sample concurrent SRDF configuration for an R1 migration. The R11 in Symmetrix A and the R2 device in Symmetrix B must be in one of the applicable pair states before issuing the `symrdf migrate -replace` command.

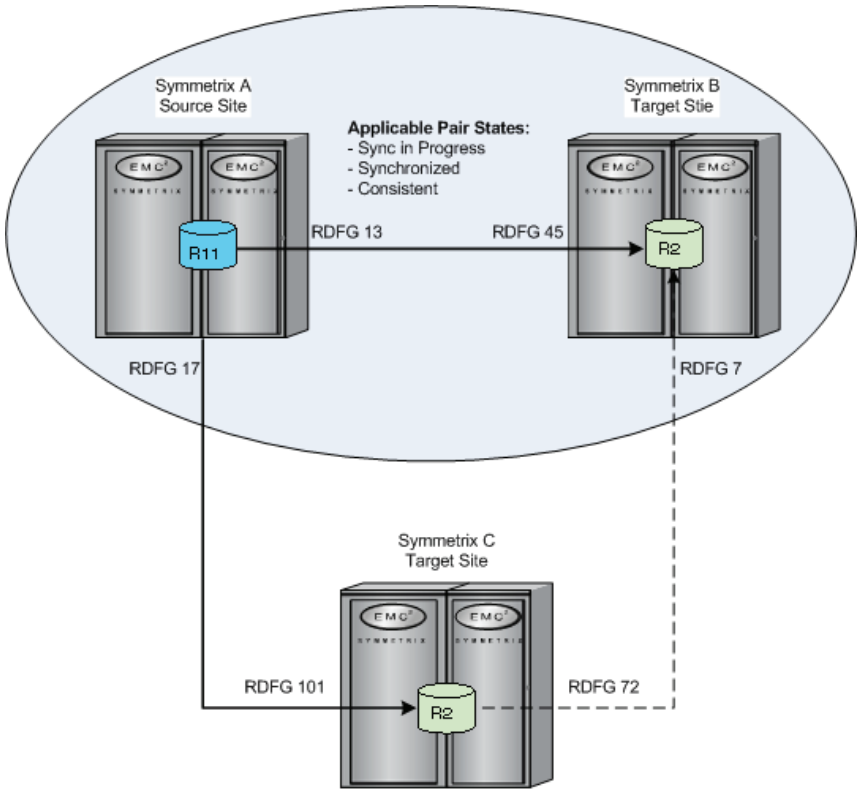


Figure 54 R1 migration: R11/R2 applicable pair states for migrate -replace (first leg)

Figure 55 shows a sample concurrent SRDF configuration for an R2 migration. The R11 in Symmetrix A and the R2 device in Symmetrix B must be in one of the states before issuing the `symrdf migrate -replace` command.

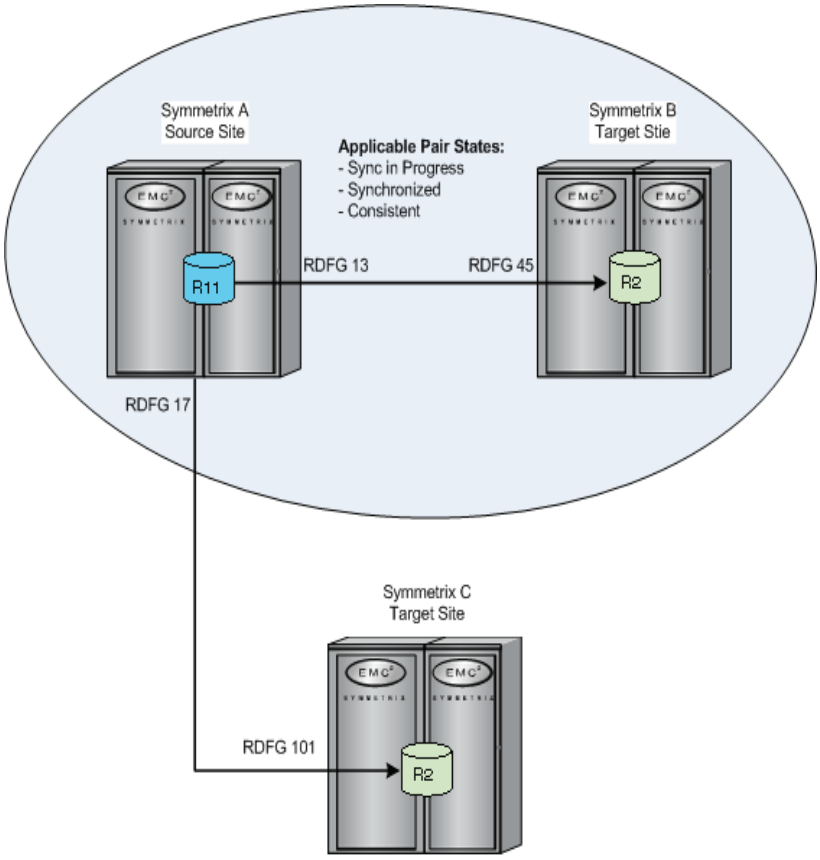


Figure 55 R2 migration:R11/R2 applicable pair states for migrate -replace (first leg)

Pair states for migrate -replace for second leg of concurrent SRDF

Before replacing an R1, the R11 and its replacement device must in a specific SRDF pair state shown in Figure 56. This temporary pairing was used to perform the concurrent SRDF data transfer to the new device. When replacing R2, the R11 and the new R2 device (new pair) must also be in a certain pair state shown in Figure 57.

Table 29 shows the applicable pair states for `symrdf migrate -replace` for an R1 and an R2 migration.

Table 29 SRDF migrate -replace control operation and applicable pair states

Control operation:	Pair state: Temporary or New ->R2											
	SYNC IN PROG	SYNCHRONIZED	SPLIT	SUSPENDED	FAILED OVER	PARTITIONED[1]	PARTITIONED[2]	R1 UPDATED	R1 UPD IN PROG	INVALID	CONSISTENT	TRANSMIT IDLE
migrate -replace	P	P									P	

1. The remote Symmetrix array is in the SYMAPI database (it was discovered).
2. The remote Symmetrix array is not in the SYMAPI database (it was not discovered or was removed).

Figure 56 shows a sample concurrent SRDF configuration for an R1 migration. The R11 device in Symmetrix A and the R2 device in Symmetrix C must be in one of the applicable pair states before issuing the `symrdf migrate -replace` command.

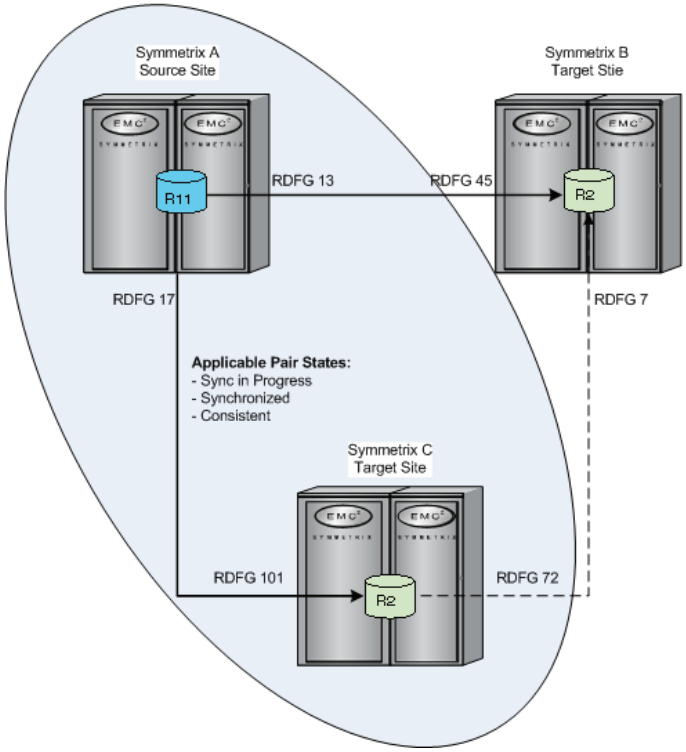


Figure 56 R1 migration: applicable R11/R2 pair states for migrate -replace (second leg)

Figure 57 shows a sample concurrent SRDF configuration for an R2 migration. The R11 in Symmetrix A and the R2 device in Symmetrix C must be in one of the states before issuing the `symrdf migrate -replace` command.

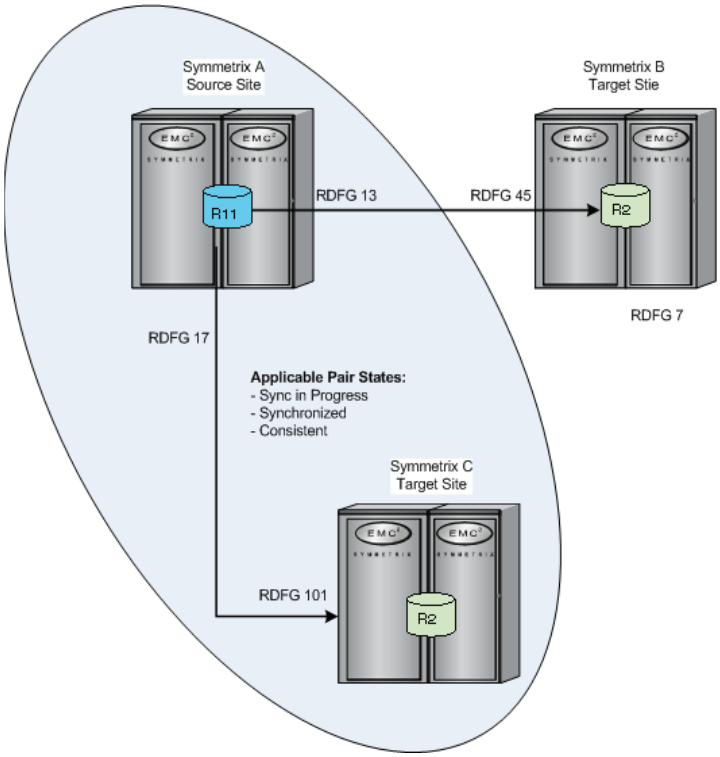


Figure 57 R2 migration: applicable R11/R2 pair states for migrate -replace (second leg)

CHAPTER 11
SRDF Automated Recovery

This chapter describes SRDF Automated Recovery responses and explains how to use the SYMCLI `symrecover` command to monitor and perform automated recovery of various SRDF/S and SRDF/A environments:

- Overview 256
- Launching SRDF Automated Recovery 259
- The symrecover options file parameters 261

Overview

SRDF Automated Recovery is a utility implemented by the `symrecover` command for optimizing ever-ready fault management responses in basic SRDF operations and cascaded SRDF environments. This utility runs in the background and monitors the state of various SRDF/S or SRDF/A sessions. If the utility detects a session failure, an automatic recovery and restart of the session is attempted, based on preconfigured settings specified in the `symrecover` options file. The options file also provides parameters for setting up error logging and event notification through email, and additional parameters for monitor, recovery strategies, and restart actions.

Note: Do not use this recovery utility in SRDF/Star environments as it is not supported.

Figure 58 shows a basic SRDF recover environment consisting of the primary R1 site replicating data to the secondary R2 site over a synchronous or asynchronous link. A gold copy (BCV or clone) of your enterprise work can be built on the R2 site to augment recovery restart strategies.

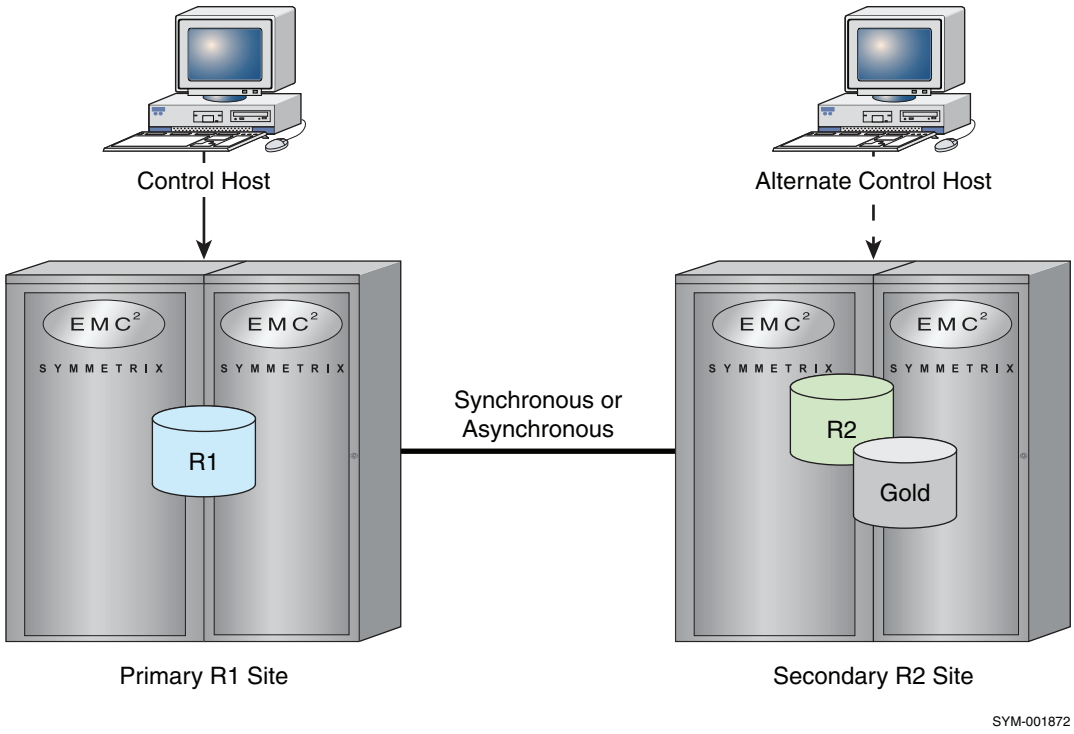


Figure 58 SRDF Recovery environment

As shown in Figure 59, the basic cascaded SRDF recovery environment consists of the primary R1 site replicating data to the secondary R21 site and replicating the same data to a tertiary R2 site. Gold copies (BCV or clone) of your enterprise work can be built on the R21 and R2 sites to augment recovery restart strategies.

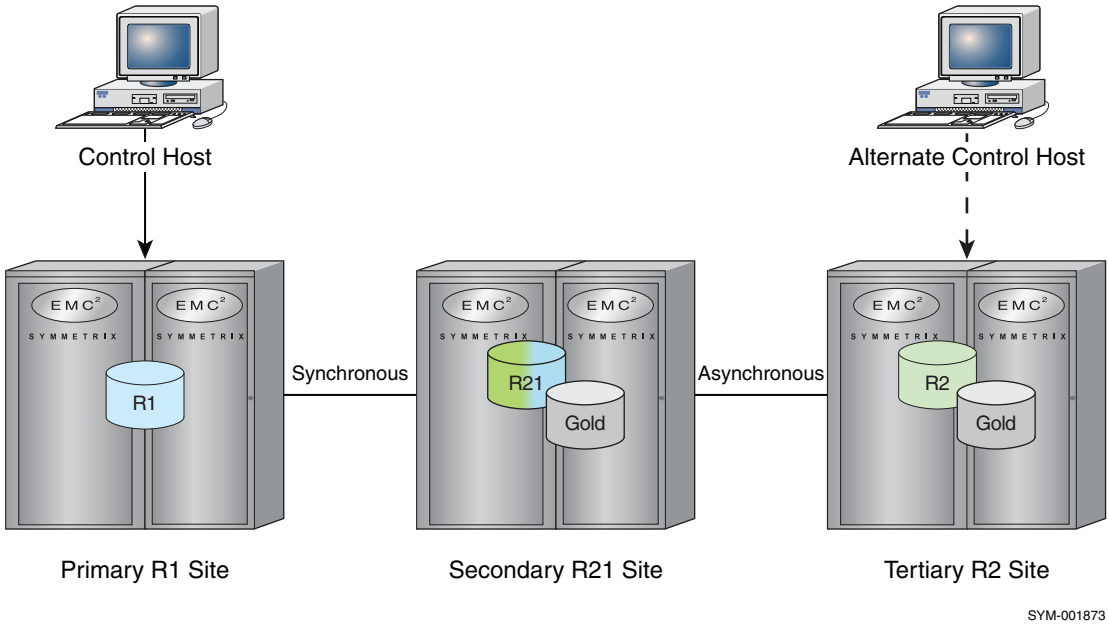


Figure 59 Cascaded SRDF recovery environment

Requirements and strategies

The following lists the general requirements for using Solutions Enabler SRDF Automated Recovery:

- The `symrecover` session must be started either at the primary R1 site or the remote R2 site.
- Solutions Enabler binaries must either be in the PATH or specified as a parameter.
- The `symrecover` command can only be run with the Perl script shipped with Solutions Enabler.
- If the group is concurrent, then `symrecover` must be run from the R1 workload site.
- The initial group state must be CONSISTENT or SYNCHRONIZED, depending on the target SRDF state, unless the `restart_group_on_startup` option is specified (not the default).

Consistency protection

The following lists restrictions and behaviors specific to consistency protection:

- If consistency protection is desired, it must be enabled prior to starting `symrecover`. A `symrecover` session must be started on the same site where consistency was enabled via a consistency group. Note that if you are managing using device groups, `symrecover` can be started at other sites.

- In cascaded SRDF environments, when managing both hops together, only sync consistency protection is allowed on the R1-to-R21 link (hop1) and only async consistency protection is allowed on the R21-to-R2 link (hop2). Here, consistency must be enabled on both hops or neither hops.
- In cascaded SRDF environments, when managing the remote R21 to R2 hop, the R1 to R21 link must be either in `Synchronized` or `Suspended` state. Otherwise, the remote R21 to R2 pair state remains in the `SyncInProg` state forever.

Gold copy needs

The following lists the restrictions/behaviors/needs specific for gold copy options:

- You can perform R2 gold copying with either Native Clones or Business Continuance Volumes (BCVs).
- BCV-to-STD association for the R2 gold copy is dynamic using the `symmir` defaults.
- For cascaded SRDF environments, native TimeFinder or TF/Clone emulation can execute from either the R21 or the remote R2 site. When you set the SYMCLI_CLONE_EMULATION parameter to enable, you override the native TF CLI default selection.
- For cascaded SRDF environments, the gold copies on both the R21 and R2 sites have to be the same type (both native TF CLI or clone emulation). The exception here would be *mixed* mode (RAID-5 or RAID-6 being employed on the same site).
- For cascaded SRDF environments, gold copying defaults to the remote R2 site. You can optionally specify to use both the R21 site and the R2 site for gold copying.

Restart behaviors

The following lists restrictions/behaviors specific to the restart procedure within the automated `symrecover` session:

- In cascaded SRDF environments, when managing both hops together, as part of the restart process, the hop2 (R21-to-R2) data synchronization starts in adaptive copy mode, as asynchronous mode may lag due to the initial high resynchronizing demand, causing the link system to choke if the cache limit is reached. Once the invalid tracks hit a value specified by the track value of the `restart_adcopy_resynch_threshold` parameter or its default of 30,000 outstanding tracks, the hop transitions to asynchronous mode and async consistency is enabled.
- In cascaded SRDF environments, when managing both hops together, a failure either on one of the hops or on both hops (when `symrecover` process looks at the state of CG/DG) triggers a single error. Then `symrecover` attempts to restart both hops together. If the `symrecover` process fails to restart both hops, it retries again until the `restart_max_attempts` parameter limit is reached.
- A recovery fails if monitoring a leg that has an R22 device when the other SRDF mirror of the R22 is read/write (RW) on the link (such states as `synchronized`, `syncinprog`, or `consistent`).
- The recovery does not start when the `-restart_group_on_starup` parameters are specified, and an R22 device has another SRDF mirror that is already RW on the link.

For a complete list of parameters and optional recovery actions to be set in the `symrecover` options file, refer to Table 30 on page 261.

The symrecover command

The SRDF Automated Recovery response is launched and optimized by the `symrecover` command, using the following command syntax:

```
symrecover [-h]
symrecover [-env | -version]

symrecover start {-g DgName | -cg CgName}
[-mode {SYNC | ASYNC}] [-out LogPath]
[-options <OptnFile>]
```

A device group or composite group must be specified, and the mode must be either sync or async.

An example command for a basic SRDF/S environment follows:

```
symrecover start -g DgName -mode sync -options OptnFile
```

Note: The symrecover command does not support the monitoring or recovery of a device group or composite group that is set with an ANY group type.

The `symrecover` command processes the command line options and options file settings.

Note: Any options specified on the command line take precedence over the options specified in the `symrecover` options file.

For a cascaded SRDF environment, the target composite group is specified (do not use the `-mode` option). The following is a sample command for a cascaded SRDF environment:

```
symrecover start -cg CgName -cascaded_monitor_both_hops -options
OptnFile
```

Note: The `symrecover` command returns an error if used with an SRDF device pair containing thin and standard devices. The thin device must be on a Symmetrix array running Enginuity 5875 or higher. The standard device must be on a Symmetrix array running Enginuity 5671, 5773.50154, or 5875 and higher (but not Enginuity 5874).

Launching SRDF Automated Recovery

The `symrecover` command launches an automated recovery response for a safe SRDF environment and can be invoked manually from the command line, but more commonly is setup to run continuously in the background by using the Windows Scheduled Tasks, the UNIX CRON/scheduled task, or a UNIX (RC.2) file.

You can run the `symrecover` command from either the R1 or the R2 side as long as all the SRDF standard devices in the device group or the composite group are local to the host. When devices in groups are not local to a host, they are marked as invalid, stop all control operations from being performed against them. You can define devices in groups on the

R2 side with a corresponding partner but `symrecover` cannot start in this environment. This is because you cannot monitor groups on the R2 side when the remote partner is concurrent. You must monitor these groups from the host.

To manually start `symrecover` for an SRDF/A composite group named `RDFAmon`, using the options file named `cg_mon_opts`, enter:

```
symrecover start -cg RDFAmon -mode async -options cg_mon_opts
```

The following example shows the option settings and their default values for a BCV gold copy in the `cg_mon_opts` file:

```
# Options file for symrecover
#######################################################
goldcopy_clone_list = TGT
goldcopy_location = R2
goldcopy_max_wait = 1800
goldcopy_resync_interval = 0
goldcopy_state_post_restart = ACTIVATED
goldcopy_state_startup = ACTIVATED
goldcopy_type = CLONE
help = 0
log_level = 3
monitor_cycle_time = 300
monitor_only = 0
out = /var/symapi/log
restart_adcopy_resynch_threshold = 30000
restart_attempt_pause = 60
restart_delay = 30
restart_group_on_startup = 0
restart_max_attempts = 5
restart_max_wait_adcopy_sync = 0
restart_max_wait_state_change = 0
restart_max_wait_warn_interval = 600
restart_rdfa_min_cycle_warn_interval = 300
restart_rdfa_min_cycle_warn_value = 0
restart_state_syncinprog_wait_time = 120
restart_state_syncinprog_warn_interval = 300
restart_state_transmit_wait_time = 120
restart_state_transmit_warn_interval = 300
restart_sync_type = ADCOPY
restart_window = 3600
run_once = 0
run_until_first_failure = 0
```

Note: If an SRDF/A group become synchronous (SRDF/S), `symrecover` attempts to reset the SRDF link to SRDF/A mode.

To recover a cascaded SRDF environment, add the following parameter settings to the previous options file:

```
cascaded_monitor_both_hops = 1
goldcopy_location = All
```

This allows recovery on both hops and builds gold copies at the R21 and R2 sites. The hop2 (R21->R2 link) also restarts quickly and safely in ADCOPY mode, during the R2 resynchronization period.

Stop symrecover

To stop `symrecover` manually, enter a Ctrl/C. To stop a `symrecover` task running in the background use one of the following options:

- Windows — Cancel the task in the Scheduled Tasks, or use End Task in the Task Manager.
- UNIX — Issue the kill command.

The symrecover options file parameters

Table 30 describes the valid settings in the `symrecover` options file.

Note: Many of these settings have Boolean values of 0 or 1. A value of 0 disables the setting and a value of 1 enables it.

Table 30 symrecover options file parameters (page 1 of 5)

Setting	Description
`cascaded_monitor_both_hops= [0|1]`	For cascaded SRDF environments, when set to 1, the `symrecover` session will monitor both hops linking the cascaded sites. When set to 1, a `symrecover` session ignores the `-mode` option and can be invoked at either the R1 primary or the remote R2 tertiary site only (not at R21 site). The default 0 setting monitors/recovers a single hop only invoked from any site.
`email_addr_target= <e_addr1, e_addr2, ..., ...>`	Email notification address on errors. If any of the `email_*` options are specified, then this option must also be specified to activate email alerts. Multiple comma delimited addresses may be specified. There is no default value.
`email_addr_source= e_addr1`	Specifies an address that will be used as the 'from' field of any e-mails that symrecover sends. No checks are done about the validity of the e-mail address. If this is not specified, then a default value is generated based on the system's hostname and current user account.
`email_server= e_srvr_addr`	Specifies the host target email server. If any of the `email_*` options are specified then this option must also be specified to activate email alerts. There is no default value.
`email_subject= err_subject_string`	Specifies the email notification subject on errors. The default value is: `SymRecover Alert: Host [HostName] Group [GrpName]`

Table 30 symrecover options file parameters (page 2 of 5)

Setting	Description
`email_log_level= SeverityLevel`	The severity level desired for the email alert triggering message. Valid values are: 0 = Off. 1 = Only Errors are reported. 2 = Errors and Warnings are reported. 3 = Errors, Warnings, and Informational messages are reported. 4 = All messages are reported, including all SYMCLI commands and responses. **Note:** For each message that meets the particular logging level requirement, an email is sent with that message. It is highly recommended that at most this be set to either a 1 or a 2. If the required email options (`email_server` and `email_addr_target`) are not specified, then the default value is 0. If they are specified then the default value is 1.
`goldcopy_location= LocationValue`	Specifies the location of the backup gold copy. Possible (case insensitive) values are: • NONE = No gold copy is desired. All other gold copy optional parameters in this list are ignored. • R2 = A gold copy on the R2 site is desired. • ALL = A gold copy on the R21 cascaded site and R2 site is desired. R2 is the default setting. Any R2 BCV pairs need to be already defined before calling `symrecover`.

`goldcopy_type=CopyType`
Old alternate, if still necessary:
`goldcopy_type_r2= CopyType`
	Specifies the type of goldcopy to create on the R2 side. Valid (case insensitive) values are:
	none = No gold copy is desired. All other `goldcopy_*` options are ignored.
	bcv = BCV gold copy on the R2 side is created, which is the default.
	clone = Clone gold copy on the R2 is created.
	Note: For the BCV gold copy, the R2 BCVs must be paired with the R2 devices before starting `symrecover`.
	For the clone gold copy, the target devices must have a clone session with the R2 devices before starting `symrecover`.

`goldcopy_state_startup= CopyType`
Old alternate, if still necessary:
`goldcopy_bcv_r2_mirror_state_startup= CopyState`
	Specifies the desired state of the R2 gold copy upon routine startup. Valid (case insensitive) values are:
	establish = the devices must be established (BCV gold copy only).
	split = The devices must be split (BCV gold copy only).
	activated = The devices must be in the copied state (clone gold copy only).
	created = The devices must be in the precopy state (clone gold copy only).
	none = The devices must be unchanged, which is the default.
	Note: If the gold copy type is BCV and the default state of the BCVs is establish, this has been shown to increase SRDF/A session drops.

Table 30 symrecover options file parameters (page 3 of 5)

Setting	Description

`goldcopy_state_post_restart= CopyState`
 	
Old alternate, if still necessary:
`goldcopy_bcv_r2_mir_state_post_restart = CopyState`
	Following a successful SRDF/A session restart or BCV resync, specifies which state the R2 gold copy should be. Valid (case insensitive) values are:
	establish = The devices must be left established (BCV gold copy only).
	split = The devices must be split, which is the default (BCV gold copy only).
	activated = The devices must be in the copied state (clone only).
	created = The devices must be in the precopy state (clone only).
	
	Note: If the gold copy type is BCV and the default state of the BCVs is establish, this has been shown to increase SRDF/A session drops.

`goldcopy_max_wait= MaxWaitTime`
 	
Old alternate, if still necessary:
`goldcopy_max_wait_bcv= MaxWaitTime`
	Specifies the length of time, in seconds, for `symrecover` to wait for synchronization. Valid values are 0 to `maxint` (2147483647). The default is 0, which indicates for `symrecover` to wait forever.
	For clone gold copies, if the `goldcopy_state_post_restart` option is set to `activated`, it waits for the clone copied state to be reached before performing synchronization. If this option is set to `created`, it waits for the clone precopied state to be reached.

`goldcopy_resync_interval= resynctime`
 	
Old alternate, if still necessary:
`goldcopy_bcv_r2_mirror_resync_interval = resynctime`
	Defines the resync interval, in minutes, for `symrecover` to automatically create a new clone gold copy or a new BCV gold copy, which overrides the existing gold copy. This action only takes place during non-error periods.
	Valid values are 0, and 15 to `maxint`. Zero (0) indicates that the mirrors are never to be automatically synchronized outside of error-producing events.The default is 15.
	
	Note: If the gold copy type is `BCV`, the act of frequently synchronizing the R2 BCVs has been shown to increase SRDF/A session drops.

`goldcopy_clone_list= List`	For a clone gold copy, this option tells `symrecover` which list within the device group or the composite group to search for clone devices. Valid (case insensitive) values are: tgt = Uses the TGT list. bcv = Uses the BCV list.
`monitor_cycle_time= cycletime`	Defines the number of seconds to pause between monitor status scans. The minimum value is 30 seconds, the maximum is 3600 seconds. The default value is 300 seconds.
`monitor_only= [0|1]`	Specifies to only monitor the state of specified group. No recovery actions will take place. This option is not enabled by default. **Note:** `monitor_only`, `run_once`, and `run_until_first_failure` are mutually exclusive options.
`run_once= [0|1]`	Specifies to check the status of the group once. If the group needs recovery actions perform them. Exit after one check. This option is not enabled by default. This option ignores the setting of `restart_max_attempts`. **Note:** `monitor_only`, `run_once`, and `run_until_first_failure` are mutually exclusive options.

Table 30 symrecover options file parameters (page 4 of 5)

Setting	Description
`run_until_first_failure= [0|1]`	Specifies to monitor the group until the first failure occurs and then exit without performing any recovery action. This option is not enabled by default. This option ignores the setting of `restart_max_attempts`. **Note:** `monitor_only`, `run_once`, and `run_until_first_failure` are mutually exclusive options.
`rdfg= rdfgvalue`	When working with device groups or composite groups that contain concurrent devices, symrecover supports monitoring only one of the SRDF groups that contain mirrors of the concurrent devices. Use the rdfg option to indicate the SRDF group that symrecover should monitor. Note that monitoring of concurrent SRDF defined groups is only supported when symrecover is executed from the R1 side. The value is taken directly as specified and no data validation is performed on it. This option is not set by default and non-concurrent SRDF groups are assumed. **Note:** If the group is a composite group, and consistency is enabled, this must be of the "name:" format and this value is case sensitive.
`restart_adcopy_resynch_threshold= tracks`	Specifies the number of tracks outstanding that during recovery will trigger a switch over to SRDF/A or SRDF/S. The default value is 30000.
`restart_attempt_pause= time`	Inserts a specified wait time before an attempt is made to restart a failed session to allow for things to settle down. After the `restart_attempt_pause` is complete, `symrecover` redrives the overall monitor loop. If there is still a problem, the restart failure count is incremented and a restart is attempted. Valid values are 30 to 3600 seconds. The default is 60 seconds.
`restart_delay= time`	Inserts a specified wait time after an attempt is made to restart a failed session and the attempt itself fails. Valid values are 0 (no delay, immediately restart) to maxint. The default is 30 seconds.
`restart_group_on_startup= [0|1]`	On `symrecover` startup, if the group being monitored is not initially in a Consistent state (for SRDF/A) or a Synchronized state (for SRDF/S), `symrecover` considers this an error condition and exits. If this option is specified, `symrecover` will attempt to recover the group on startup. This option is not enabled by default.
`restart_max_attempts= attempts`	Specifies the maximum number of restart attempts that are performed within the `restart_window` interval. After this limit is reached the program will terminate. The range is from 0 to maxint. The value of 0 means to attempt indefinitely.The default is 5 attempts.
`restart_max_wait_adcopy_sync= time`	Specifies the length of time (in seconds) during a restart for a program to wait for a group to achieve the `restart_adcopy_resync_threshold` number of tracks pending. Valid values are 0 to maxint. The value of 0 means to infinitely wait. The default is 0.
`restart_max_wait_state_change= statetime`	Specifies the length of time (in seconds) during a restart for a program to wait for a group to change to a desired state (once requested). Valid values are 0 to maxint. The value of 0 means to infinitely wait. The default is 0.

Table 30 symrecover options file parameters (page 5 of 5)

Setting	Description
`restart_max_wait_warn_interval= warntime`	Specifies the length of time (in seconds) during a restart, while waiting for a state change to occur, to display a progress warning message. Valid values are 0 and 30 to maxint. The value of 0 means to wait forever. The default is 600 seconds.
`restart_rdfa_min_cycle_warn_interval= cyclewarntime`	Specifies the length of time (in seconds) before repetitively displaying a warning when the RDFA minimum cycle time exceeds the `restart_rdfa_min_cycle_warn_value` parameter. Valid values are 30 to `maxint`.The default is 600.
`restart_rdfa_min_cycle_warn_value= warntime`	Specifies the maximum value (in seconds) to which a trigger can occur with a warning message, indicating the RDFA minimum cycle time has exceeded this value. Valid values are 0 and 30 to `maxint`. The value of 0 means this feature is turned off, which is the default.
`restart_state_syncinprog_wait_time time`	The maximum length of time (in seconds) during a group syncinprog state that sleep is done before rechecking the group status. Valid values are [30] to [maxint]. The default is [120] seconds.
`restart_state_transmit_warn_interval= time`	Specifies the interval of time (in seconds) that while a group remains in a transmit idle state, to generate a warning message. Valid values are 0 to `maxint`. The default is 300 seconds.
`restart_state_transmit_wait_time= transwaittime`	Specifies the maximum length of time (in seconds) that during a group transmit idle state, a sleep is done before rechecking the group status. Valid values are 30 to maxint. The default is 120 seconds.
`restart_sync_type= synctype`	Specifies the type of synchronization to be used following the detection of a failed SRDF/A session. Valid values are: ADCOPY = adaptive copy disk (default). SYNC = synchronous mode. NONE = No intermediate track resynch stage will be attempted. A direct re-establish using the existing SRDF session mode will be attempted. Note that if `cascaded_monitor_both_hops` is set, `restart_sync_type` is ignored as ADCOPY is used in the R21->R2 link at restart.
`restart_window= time`	Specifies a time window (in seconds) during which no more than restart_max_attempts failures and accompanying restart attempts will be tolerated before monitoring is terminated. The window begins at the time of the first failure and ends restart_window seconds later. A new window begins with a failure after expiration of the previous window.
`log_level= level`	The desired logging level. Valid values are: 0 = Off. 1 = Only Errors are reported. 2 = Errors and Warnings are reported. 3 = Errors, Warnings, and Informational messages are reported (default). 4 = All messages are reported.

PART 2
Operational Examples

Part 2 of this guide describes typical SRDF tasks. These taks represent common practices in the management of your Symmetrix storage environment.

Each chapter includes multiple practical examples. These examples show the SYMCLI command sequences to accomplish the specific task. The tasks described in Part 2 include:

Chapter 13, "Performing SRDF Control Operations,"

Managing devices in various SRDF configurations using SRDF control operations.

Chapter 14, "Querying and Verifying with SRDF Commands,"

Using the query and verify operations with SRDF family products.

Chapter 15, "Implementing Consistency Protection,"

Implementing consistency protection across one or more database management systems within an SRDF configuration using SRDF Enginuity Consistency Assist (RDF-ECA) for synchronous mode and the SRDF Multi Session Consistency (MSC) for asynchronous mode.

Chapter 16, "Performing SRDF/Automated Replication Operations,"

Replicating data in pre-defined cycles using the SRDF automated replication process.

Some of the examples in this section were performed with lower versions of the software. Therefore, your output may not look exactly like the output appearing in these examples.

CHAPTER 13
Performing SRDF Control Operations

This chapter contains the following examples of SRDF control operations:

- Example 1: Basic SRDF control operations 270
- Example 2: Concurrent SRDF 292
- Example 3: Creating a dynamic SRDF group 311
- Example 4: Creating dynamic SRDF pairs 314
- Example 5: Operating with SRDF asynchronous replication 321
- Example 6: Using a composite group to control SRDF pairs 329
- Example 7: Creating concurrent dynamic SRDF pairs 337
- Example 8: Failing over cascaded SRDF 342

Note: The commands in this chapter were executed using Solutions Enabler V7.2.

Example 1: Basic SRDF control operations

The hardware setup in this example consists of two hosts, one connected to a local (source) Symmetrix array and the other connected to a remote (target) Symmetrix array.

- The following `symrdf list` command displays information about the local (R1) and remote (R2) SRDF devices. The RDF Typ:G column identifies a device as R1 or R2 and its SRDF group number, which is shown after the colon.

```
‹R1Host#› symrdf list -sid 321

Symmetrix ID: 000192600321

                              Local Device View
---------------------------------------------------------------------------
                     STATUS     MODES                       RDF  S T A T E S
Sym        RDF       ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G     SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------   --------- ----- -------  ------- --- ---- -------------

0060 0060   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0061 0061   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0062 0062   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0063 0063   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0064 0064   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0065 0065   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0066 0066   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0067 0067   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0250 0250 B-R2:35    ?? WD RW   S..2.       0        0 WD  RW   Synchronized
07E0 07E0   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E1 07E1   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E2 07E2   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E3 07E3   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E4 07E4   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E5 07E5   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E6 07E6   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E7 07E7   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E8 07E8   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07E9 07E9   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07EA 07EA   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07EB 07EB   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07EC 07EC   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07ED 07ED   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07EE 07EE   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
07EF 07EF   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0800 0800   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0801 0801   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0802 0802   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0803 0803   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0808 0808   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0809 0809   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080A 080A   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080B 080B   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080C 080C   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080D 080D   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080E 080E   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
080F 080F   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0810 0810   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0811 0811   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0812 0812   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0813 0813   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0814 0814   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0815 0815   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
0816 0816   R2:129   RW WD RW   S..2.       0        0 WD  RW   Synchronized
```

```
0817 0817   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0828 0828   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0829 0829   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082A 082A   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082B 082B   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082C 082C   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082D 082D   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082E 082E   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
082F 082F   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0830 0830   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0831 0831   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0832 0832   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0833 0833   R2:129  RW WD RW   S..2.       0        0 WD  RW   Synchronized

Total                                 -------- --------
  Track(s)                                   0        0
  MB(s)                                    0.0      0.0
```

- The `symdev list` command with the `-r1` option displays all R1 devices. Any R1 devices not belonging to a device group is displayed as N/Grp'd, and can be added to an RDF1 device group.

‹R1Host#› **symdev list -sid 321 -r1**

```
Symmetrix ID: 000192600321

        Device Name           Directors                  Device
--------------------------- ------------- -------------------------------------
                                                                          Cap
Sym  Physical               SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- -------------------------------------

0060 /dev/rhdisk27          10E:0 10A:D0  RDF1+Mir      N/Grp'd      RW    4314
0061 /dev/rhdisk28          10E:0 09C:C0  RDF1+Mir      N/Grp'd      RW    4314
0062 /dev/rhdisk29          10E:0 10B:C0  RDF1+Mir      N/Grp'd      RW    4314
0063 /dev/rhdisk30          10E:0 09D:D0  RDF1+Mir      N/Grp'd      RW    4314
0064 /dev/rhdisk31          10E:0 09A:C2  RDF1+Mir      N/Grp'd      RW    4314
0065 /dev/rhdisk32          10E:0 10C:D2  RDF1+Mir      N/Grp'd      RW    4314
0066 /dev/rhdisk33          10E:0 09B:D2  RDF1+Mir      N/Grp'd      RW    4314
0067 /dev/rhdisk34          10E:0 10D:C2  RDF1+Mir      N/Grp'd      RW    4314
07E0 /dev/rhdisk43          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E1 /dev/rhdisk44          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E2 /dev/rhdisk45          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E3 /dev/rhdisk46          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E4 /dev/rhdisk47          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E5 /dev/rhdisk48          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E6 /dev/rhdisk49          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E7 /dev/rhdisk50          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E8 /dev/rhdisk51          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07E9 /dev/rhdisk52          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07EA /dev/rhdisk53          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07EB /dev/rhdisk54          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07EC /dev/rhdisk55          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07ED /dev/rhdisk56          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07EE /dev/rhdisk57          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
07EF /dev/rhdisk58          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
0800 /dev/rhdisk76          10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
0801 /dev/rhdisk216         10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
0802 /dev/rhdisk217         10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
0803 /dev/rhdisk218         10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
```

Creating a device group and adding devices to it are prerequisites for performing SRDF operations. In the following example:

- The `symdg create` command creates the `Rdf1Grp` device group.

- The `symdg addall` command adds all devices belonging to SRDF group 128 to `Rdf1Grp` on Symmetrix 321:

```
‹R1Host#› symdg create Rdf1Grp -type rdf1
‹R1Host#› symdg -g Rdf1Grp -sid 321 addall dev -sel_rdfg 128 -v

Device(s) 00060:00067: ADD DEVS BY RANGE Succeeded.

Device(s) 007E0:007EF: ADD DEVS BY RANGE Succeeded.

Device(s) 00800:00803: ADD DEVS BY RANGE Succeeded.
```

- The `symdg show` command displays information applicable to all SRDF standard devices in device group `Rdf1Grp` (scroll to `Device Group RDF Information` in the example:

```
‹R1Host#› symdg show Rdf1Grp

Group Name:  Rdf1Grp

    Group Type                                    : RDF1     (RDFA)
    Device Group in GNS                           : Yes
    Valid                                         : Yes
    Symmetrix ID                                  : 000192600321
    Group Creation Time                           : Tue Jul 20 10:10:53 2010
    Vendor ID                                     : EMC Corp
    Application ID                                : SYMCLI

    Number of STD Devices in Group                :   28
    Number of Associated GK's                     :    0
    Number of Locally-associated BCV's            :    0
    Number of Locally-associated VDEV's           :    0
    Number of Locally-associated TGT's            :    0
    Number of Remotely-associated VDEV's(STD RDF):    0
    Number of Remotely-associated BCV's (STD RDF):    0
    Number of Remotely-associated TGT's(TGT RDF) :    0
    Number of Remotely-associated BCV's (BCV RDF):    0
    Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
    Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
    Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
    Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0
    Number of Composite Groups                    :    0
    Composite Group Names                         : N/A

    Standard (STD) Devices (28):
        {
        --------------------------------------------------------------------------------
                                                  Sym  Device                      Cap
        LdevName              PdevName            Dev  Config        Att. Sts     (MB)
        --------------------------------------------------------------------------------
        DEV001                /dev/rhdisk27       0060 RDF1+Mir           RW      4314
        DEV002                /dev/rhdisk28       0061 RDF1+Mir           RW      4314
        DEV003                /dev/rhdisk29       0062 RDF1+Mir           RW      4314
        DEV004                /dev/rhdisk30       0063 RDF1+Mir           RW      4314
        DEV005                /dev/rhdisk31       0064 RDF1+Mir           RW      4314
        DEV006                /dev/rhdisk32       0065 RDF1+Mir           RW      4314
        DEV007                /dev/rhdisk33       0066 RDF1+Mir           RW      4314
        DEV008                /dev/rhdisk34       0067 RDF1+Mir           RW      4314
        DEV009                /dev/rhdisk43       07E0 RDF1+TDEV          RW      4314
        DEV010                /dev/rhdisk44       07E1 RDF1+TDEV          RW      4314
        DEV011                /dev/rhdisk45       07E2 RDF1+TDEV          RW      4314
        DEV012                /dev/rhdisk46       07E3 RDF1+TDEV          RW      4314
        DEV013                /dev/rhdisk47       07E4 RDF1+TDEV          RW      4314
        DEV014                /dev/rhdisk48       07E5 RDF1+TDEV          RW      4314
        DEV015                /dev/rhdisk49       07E6 RDF1+TDEV          RW      4314
```

```
    DEV016                /dev/rhdisk50          07E7 RDF1+TDEV          RW      4314
    DEV017                /dev/rhdisk51          07E8 RDF1+TDEV          RW      4314
    DEV018                /dev/rhdisk52          07E9 RDF1+TDEV          RW      4314
    DEV019                /dev/rhdisk53          07EA RDF1+TDEV          RW      4314
    DEV020                /dev/rhdisk54          07EB RDF1+TDEV          RW      4314
    DEV021                /dev/rhdisk55          07EC RDF1+TDEV          RW      4314
    DEV022                /dev/rhdisk56          07ED RDF1+TDEV          RW      4314
    DEV023                /dev/rhdisk57          07EE RDF1+TDEV          RW      4314
    DEV024                /dev/rhdisk58          07EF RDF1+TDEV          RW      4314
    DEV025                /dev/rhdisk76          0800 RDF1+TDEV          RW      4314
    DEV026                /dev/rhdisk216         0801 RDF1+TDEV          RW      4314
    DEV027                /dev/rhdisk217         0802 RDF1+TDEV          RW      4314
    DEV028                /dev/rhdisk218         0803 RDF1+TDEV          RW      4314
    }
```

Device Group RDF Information

```
    {
    RDF Type                                : R1
    RDF (RA) Group Number                   : 128               (7F)

    Remote Symmetrix ID                     : 000192600256

    R2 Device Is Larger Than The R1 Device : False

    Paired with a Diskless Device           : False
    Paired with a Concurrent Device         : False
    Paired with a Cascaded Device           : False
    Thick Thin Relationship                 : False

    RDF Pair Configuration                  : Normal
    RDF STAR Mode                           : False

    RDF Mode                                : Synchronous
    RDF Adaptive Copy                       : Disabled
    RDF Adaptive Copy Write Pending State   : N/A
    RDF Adaptive Copy Skew (Tracks)         : 32767

    RDF Device Domino                       : Disabled

    RDF Link Configuration                  : Fibre
    RDF Link Domino                         : Disabled
    Prevent Automatic RDF Link Recovery     : Enabled
    Prevent RAs Online Upon Power ON        : Enabled

    Device RDF Status                       : Ready           (RW)

    Device RA Status                        : Ready           (RW)
    Device Link Status                      : Ready           (RW)
    Time of Last Device Link Status Change : N/A

    Device Suspend State                    : N/A
    Device Consistency State                : Disabled
    Device Consistency Exempt State         : Disabled
    RDF R2 Not Ready If Invalid             : Disabled

    Device RDF State                        : Ready           (RW)
    Remote Device RDF State                 : Write Disabled  (WD)

    RDF Pair State (  R1 <===> R2 )         : Synchronized

    Number of R1 Invalid Tracks             : 0
    Number of R2 Invalid Tracks             : 0

    RDFA Information:
        {
        Session Number                      : 127
```

```
            Cycle Number                        : 0
            Number of Devices in the Session    : 28
            Session Status                      : Inactive
            Consistency Exempt Devices          : No

            Session Consistency State           : N/A
            Minimum Cycle Time                  : 00:00:30
            Average Cycle Time                  : 00:00:00
            Duration of Last cycle              : 00:00:00
            Session Priority                    : 33

            Tracks not Committed to the R2 Side: 0
            Time that R2 is behind R1           : 00:00:00
            R2 Image Capture Time               : N/A
            R2 Data is Consistent               : N/A
            R1 Side Percent Cache In Use        :  0
            R2 Side Percent Cache In Use        :  0

            Transmit Idle Time                  : 00:00:00
            R1 Side DSE Used Tracks             :  0
            R2 Side DSE Used Tracks             :  0
            R1 Side Shared Tracks               : 0
            }
        }
```

When EMC installs an SRDF configuration, the installers usually establish static SRDF pairs at that time.

The following query shows the state of the SRDF devices and their SRDF links. Under normal circumstances, the SRDF pair is Synchronized (as shown below).

The R1 devices and the SRDF links are read writable (RW). However, the R2 devices, which are acting as mirrors to the R1 devices, are write disabled (WD), and cannot be written to by the remote host at this time. The link is operating with synchronous replication (indicated by an S in the M column).

‹R1Host#› **symrdf -g Rdf1Grp query**

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
```

```
MB(s)              0.0      0.0                 0.0      0.0
```

The `symcfg list -ra all` command helps you examine Symmetrix array connections.

Executed from the local host, this command reaches all Symmetrix arrays (one or two hops away) accessible through SRDF links, and displays the remote link director information.

The Remote SymmID column shows that all three Symmetrix arrays (321, 198 and 256) are connected to each other.

```
‹R1Host#› symcfg list -ra all

Symmetrix ID: 000192600321 (Local)

                 S Y M M E T R I X    R D F    D I R E C T O R S

                                          Remote        Local    Remote
Ident  Symb  Num  Slot  Type       Attr  SymmID        RA Grp   RA Grp  Status

RF-7F   07F   87     7  RDF-BI-DIR   -   000192600256 128 (7F) 128 (7F) Online
                                       -   000192600198  12 (0B)  12 (0B)
                                       -   000192600256  13 (0C)  13 (0C)
                                       -   000192600198  12 (0B)  12 (0B)
                                       -   000192600256  13 (0C)  13 (0C)
                                       -   000192600256  16 (0F)  16 (0F)
                                       -   000192600198  15 (0E)  15 (0E)
                                       -   000192600198  15 (0E)  15 (0E)
                                       -   000192600256  16 (0F)  16 (0F)
                                       -   000192600256 128 (7F) 128 (7F)
                                       -   000192600256  35 (22)  35 (22)
                                       -   000192600198 111 (6E) 111 (6E)
                                       -   000192600198 228 (E3) 228 (E3)
                                       -   000192600198 228 (E3) 228 (E3)
RF-8F   08F   88     8  RDF-BI-DIR   -   000192600198  12 (0B)  12 (0B) Online
                                       -   000192600256 128 (7F) 128 (7F)
                                       -   000192600198  12 (0B)  12 (0B)
                                       -   000192600256  13 (0C)  13 (0C)
                                       -   000192600256  13 (0C)  13 (0C)
                                       -   000192600198  15 (0E)  15 (0E)
                                       -   000192600198  15 (0E)  15 (0E)
                                       -   000192600256  16 (0F)  16 (0F)
                                       -   000192600256  16 (0F)  16 (0F)
                                       -   000192600256 128 (7F) 128 (7F)
                                       -   000192600198 228 (E3) 228 (E3)
                                       -   000192600198 228 (E3) 228 (E3)
                                       -   000192600198 110 (6D) 110 (6D)
RF-9F   09F   89     9  RDF-R2       -   000192600256 104 (67) 102 (65) Online
                                       -   000192600256 129 (80) 129 (80)
                                       -   000192600198 229 (E4) 229 (E4)
                                       -   000192600256 129 (80) 129 (80)
                                       -   000192600198 229 (E4) 229 (E4)
RF-10F  10F   90    10  RDF-R2       -   000192600256 129 (80) 129 (80) Online
                                       -   000192600198 229 (E4) 229 (E4)
                                       -   000192600256 105 (68) 103 (66)
                                       -   000192600198 229 (E4) 229 (E4)
                                       -   000192600256 129 (80) 129 (80)
RE-7G   07G  103     7  RDF-R1       -   -              -        -      Online
RE-8G   08G  104     8  RDF-R1       -   -              -        -      Online
RE-7H   07H  119     7  RDF-R1       -   -              -        -      Online
RE-8H   08H  120     8  RDF-R1       -   -              -        -      Online

Symmetrix ID: 000192600198 (Remote)

                 S Y M M E T R I X    R D F    D I R E C T O R S
```

```
                                                Remote        Local    Remote
Ident  Symb  Num  Slot  Type        Attr  SymmID        RA Grp   RA Grp  Status

RF-7F   07F   87     7  RDF-BI-DIR   -   000192600284  17 (10)  17 (10) Online
                                     -   000192600284  17 (10)  17 (10)
                                     -   000192600321  12 (0B)  12 (0B)
                                     -   000192600321  12 (0B)  12 (0B)
                                     -   000192600256  14 (0D)  14 (0D)
                                     -   000192600256 178 (B1) 178 (B1)
                                     -   000192600321  15 (0E)  15 (0E)
                                     -   000192600321  15 (0E)  15 (0E)
                                     -   000192600321 228 (E3) 228 (E3)
                                     -   000192600321 228 (E3) 228 (E3)
                                     -   000192600256 178 (B1) 178 (B1)
                                     -   000192600256  11 (0A)  11 (0A)
                                     -   000192600256  14 (0D)  14 (0D)
                                     -   000192600256  90 (59) 100 (63)
                                     -   000192600256  11 (0A)  11 (0A)
RF-8F   08F   88     8  RDF-BI-DIR   -   000192600284  17 (10)  17 (10) Online
                                     -   000192600284  17 (10)  17 (10)
                                     -   000192600256 178 (B1) 178 (B1)
                                     -   000192600321  12 (0B)  12 (0B)
                                     -   000192600321  12 (0B)  12 (0B)
                                     -   000192600256  11 (0A)  11 (0A)
                                     -   000192600256  14 (0D)  14 (0D)
                                     -   000192600321  15 (0E)  15 (0E)
                                     -   000192600321  15 (0E)  15 (0E)
                                     -   000192600321 228 (E3) 228 (E3)
                                     -   000192600321 228 (E3) 228 (E3)
                                     -   000192600256  11 (0A)  11 (0A)
                                     -   000192600256  14 (0D)  14 (0D)
                                     -   000192600321 110 (6D) 110 (6D)
                                     -   000192600256  33 (20)  33 (20)
                                     -   000192600256 178 (B1) 178 (B1)
RF-9F   09F   89     9  RDF-R2       -   000192600321 229 (E4) 229 (E4) Online
                                     -   000192600321 229 (E4) 229 (E4)
                                     -   000192600256  91 (5A) 101 (64)
                                     -   000192600256 179 (B2) 179 (B2)
                                     -   000192600256 179 (B2) 179 (B2)
RF-10F  10F   90    10  RDF-BI-DIR   -   000192600321 229 (E4) 229 (E4) Online
                                     -   000192600321 229 (E4) 229 (E4)
                                     -   000192600256 179 (B2) 179 (B2)
                                     -   000192600321 111 (6E) 111 (6E)
                                     -   000192600256 179 (B2) 179 (B2)
RE-7G   07G  103     7  RDF-R1       -   -              -        -      Online
RE-8G   08G  104     8  RDF-R1       -   -              -        -      Online
RF-7H   07H  119     7  RDF-R1       -   -              -        -      Online
RE-8H   08H  120     8  RDF-R1       -   -              -        -      Online

Symmetrix ID: 000192600256 (Remote)

                 S Y M M E T R I X   R D F   D I R E C T O R S


                                                Remote        Local    Remote
Ident  Symb  Num  Slot  Type        Attr  SymmID        RA Grp   RA Grp  Status

RF-7F   07F   87     7  RDF-BI-DIR   -   000192600321 128 (7F) 128 (7F) Online
                                     -   000192600198 178 (B1) 178 (B1)
                                     -   000192600198  11 (0A)  11 (0A)
                                     -   000192600198 178 (B1) 178 (B1)
                                     -   000192600198  14 (0D)  14 (0D)
                                     -   000192600198  11 (0A)  11 (0A)
                                     -   000192600198  14 (0D)  14 (0D)
                                     -   000192600321  13 (0C)  13 (0C)
                                     -   000192600321  16 (0F)  16 (0F)
```

```
                                        -   000192600321  35 (22)  35 (22)
                                        -   000192600321 128 (7F) 128 (7F)
                                        -   000192600321  13 (0C)  13 (0C)
                                        -   000192600321  16 (0F)  16 (0F)
                                        -   000192600198 100 (63)  90 (59)
RF-8F   08F   88     8  RDF-BI-DIR      -   000192600198  11 (0A)  11 (0A) Online
                                        -   000192600198 178 (B1) 178 (B1)
                                        -   000192600321  13 (0C)  13 (0C)
                                        -   000192600198  11 (0A)  11 (0A)
                                        -   000192600198  14 (0D)  14 (0D)
                                        -   000192600198  33 (20)  33 (20)
                                        -   000192600198 178 (B1) 178 (B1)
                                        -   000192600321  16 (0F)  16 (0F)
                                        -   000192600321 128 (7F) 128 (7F)
                                        -   000192600321  13 (0C)  13 (0C)
                                        -   000192600321  16 (0F)  16 (0F)
                                        -   000192600321 128 (7F) 128 (7F)
                                        -   000192600198  14 (0D)  14 (0D)
RF-9F   09F   89     9  RDF-R1          -   000192600198 101 (64)  91 (5A) Online
                                        -   000192600198 179 (B2) 179 (B2)
                                        -   000192600198 179 (B2) 179 (B2)
                                        -   000192600321 102 (65) 104 (67)
                                        -   000192600321 129 (80) 129 (80)
                                        -   000192600321 129 (80) 129 (80)
RF-10F  10F   90    10  RDF-R1          -   000192600198 179 (B2) 179 (B2) Online
                                        -   000192600321 129 (80) 129 (80)
                                        -   000192600198 179 (B2) 179 (B2)
                                        -   000192600321 103 (66) 105 (68)
                                        -   000192600321 129 (80) 129 (80)
RE-7G   07G  103     7  RDF-R1          -   -              -        -      Online
RE-8G   08G  104     8  RDF-R1          -   -              -        -      Online
RE-7H   07H  119     7  RDF-R1          -   -              -        -      Online
RE-8H   08H  120     8  RDF-R1          -   -              -        -      Online
```

At the remote host, the target Symmetrix 256 is the source array.

The `symcfg list` command shows Symmetrix 256 as local and Symmetrix 198 as remote:

‹R2Host#› **symcfg list**

```
                              S Y M M E T R I X

                                        Mcode    Cache      Num Phys  Num Symm
    SymmID       Attachment  Model      Version  Size (MB)  Devices   Devices

    000192600256 Local       VMAX-1     5875       24576       145      2953
    000192600321 Local       VMAX-1     5875       24576         1      2953
    000192600198 Remote      VMAX-1     5875       24576         0      2953
```

At the remote host, the `symrdf list` command displays the local devices in the Sym Dev column and the remote devices in the RDev column.

The B-R1:35 value for device 250 in the RDF Typ:G column indicates that it is an SRDF BCV (-B) device of RDF1 (R1) type belonging to RDF group 35.

‹R2Host#› **symrdf list -sid 256**

```
Symmetrix ID: 000192600256

                            Local Device View
---------------------------------------------------------------------------
                    STATUS     MODES                      RDF  S T A T E S
Sym         RDF     ---------  -----  R1 Inv   R2 Inv ----------------------
```

```
Dev  RDev  Typ:G    SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------  --------- ----- -------  ------- --- ---- -------------

0060 0060   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0061 0061   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0062 0062   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0063 0063   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0064 0064   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0065 0065   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0066 0066   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0067 0067   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0250 0250 B-R1:35   ?? RW RW   S..1.       0        0 RW  WD   Synchronized
07E0 07E0   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
07E1 07E1   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
07E2 07E2   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
07E3 07E3   R2:128  RW WD RW   S..2.       0        0 WD  RW   Synchronized
.
.
.
.
0830 0830   R1:129  RW RW RW   S..1.       0        0 RW  WD   Synchronized
0831 0831   R1:129  RW RW RW   S..1.       0        0 RW  WD   Synchronized
0832 0832   R1:129  RW RW RW   S..1.       0        0 RW  WD   Synchronized
0833 0833   R1:129  RW RW RW   S..1.       0        0 RW  WD   Synchronized

Total                                 -------- --------
  Track(s)                                   0        0
  MB(s)                                    0.0      0.0
```

To issue the same SRDF commands from the remote host as from the local host, you must build an RDF2 remote-site device group with the same definitions as the RDF1 local-site device group.

In the following example, the `symdg export` command creates a text file (`Rdf1Grp.txt`) that contains the RDF1 group definitions.

You then use `rcp` (or `ftp`) to transfer this file to the remote host.

```
symdg export Rdf1Grp -f Rdf1Grp.txt -rdf
rcp Rdf1Grp.txt
```

- At the remote host, use the `symdg import` command to build the RDF2 device group using the definitions from the specified file (`Rdf1Grp.txt` in the following example):

```
‹R2Host#› symdg import Rdf2Grp -f Rdf1Grp.txt

Adding STD device 0060 as DEV001...
Adding STD device 0061 as DEV002...
Adding STD device 0062 as DEV003...
.
.
.
.
Adding STD device 0801 as DEV026...
Adding STD device 0802 as DEV027...
Adding STD device 0803 as DEV028...
```

- Use the `symdg list ld` command to display the contents of the new `Rdf2Grp` device group:

```
‹R2Host#› symdg -g Rdf2Grp list ld

Device Group (DG) Name: Rdf2Grp
DG's Type             : RDF2
DG's Symmetrix ID     : 000192600256
```

```
        Standard Device Name           Directors              Device
----------------------------------   ------------- ----------------------------
                                                                          Cap
Logical     Physical          Sym    SA :P DA :IT  Config        Att Sts   (MB)
----------------------------------   ------------- ----------------------------

DEV001      /dev/rhdisk148    0060   10E:0 10A:D0  RDF2+Mir          WD    4314
DEV002      /dev/rhdisk149    0061   10E:0 09C:C0  RDF2+Mir          WD    4314
DEV003      /dev/rhdisk150    0062   10E:0 10B:C0  RDF2+Mir          WD    4314
.
.
.
.
DEV026      /dev/rhdisk198    0801   10E:0  NA:NA  RDF2+TDEV         WD    4314
DEV027      /dev/rhdisk199    0802   10E:0  NA:NA  RDF2+TDEV         WD    4314
DEV028      /dev/rhdisk200    0803   10E:0  NA:NA  RDF2+TDEV         WD    4314
```

- At the remote host, the following query displays the status of the `Rdf2Grp` device group.

  The link is operating in synchronous replication, and the STATE of the R2 devices is write disabled (WD).

‹R2Host#› **symrdf -g Rdf2Grp query**

```
Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                   LI      ST
Standard      A                    N       A
Logical       T  R1 Inv   R2 Inv   K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks   S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 RW 0060 RW       0        0 S...  Synchronized
DEV002  0061 WD       0        0 RW 0061 RW       0        0 S...  Synchronized
DEV003  0062 WD       0        0 RW 0062 RW       0        0 S...  Synchronized
.
.
.
.
DEV026  0801 WD       0        0 RW 0801 RW       0        0 S...  Synchronized
DEV027  0802 WD       0        0 RW 0802 RW       0        0 S...  Synchronized
DEV028  0803 WD       0        0 RW 0803 RW       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- In the following sequence, commands do the following:
  - Create a volume group called `volgrprdf1` on the `hdisk27` device.
  - Activate the volume group.
  - Create the `R1-1` file system.
  - Mount `R1-1`.

- Create `firstfile` on `R1-1`.
- List the contents of `R1-1`.
- Unmount `R1-1`.

```
‹R1Host#› mkvg -f -y volgrprdf1 hdisk27
‹R1Host#› varyonvg volgrprdf1
‹R1Host#› crfs -v jfs2 -g volgrprdf1 -m /R1-1 -a size=100M

File system created successfully.
106288 kilobytes total disk space.
New File System size is 212992

‹R1Host#› mount /R1-1
‹R1Host#› touch /R1-1/firstfile
‹R1Host#› ls -l /R1-1

total 0
-rw-r--r--    1 root     system           0 Jul 20 13:06 firstfile
drwxr-xr-x    2 root     system         256 Jul 20 13:05 lost+found

‹R1Host#› umount /R1-1
‹R1Host#›varyoffvg volgrprdf1
‹R1Host#›exportvg volgrprdf1
```

◆ At the local host, the `symrdf split` command splits the SRDF pairs in the device group.

As part of the `symrdf split` command, the individual operations suspend and `rw_enable r2` is performed.

When the `split` is complete, a query shows the altered state of the links and the R2 devices.

```
‹R1Host#› symrdf -g Rdf1Grp split -noprompt

An RDF 'Split' operation execution is
in progress for device group 'Rdf1Grp'. Please wait...

    Suspend RDF link(s).....................................Done.
    Read/Write Enable device(s) on RA at target (R2)........Done.

The RDF 'Split' operation successfully executed for
device group 'Rdf1Grp'.
```

◆ The following query shows the links were logically set to NR (not ready) and the state of the R2 devices was changed from WD to RW:

```
‹R1Host#› symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
              ST                  LI      ST
Standard       A                   N       A
Logical        T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device   Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
```

```
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 RW       0        0 S...  Split
DEV002  0061 RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 NR 0062 RW       0        0 S...  Split
.
.
.
.
DEV026  0801 RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following commands enable the R1-1 file system to be copied to the R2-1 file system on the remote host:

```
‹R2Host#›mkdir /R2-1
‹R2Host#›importvg -y volgrprdf2 hdisk148
‹R2Host#›mount /dev/fslv00 /R2-1
‹R2Host#›ls -l /R2-1

total 0
-rw-r--r--    1 root     system            0 Jul 20 13:06 firstfile
drwxr-xr-x    2 root     system          256 Jul 20 13:05 lost+found
```

- While the local and remote sites are split, both R1 and R2 devices are accessible for reads and writes.

  The following commands (executed at the remote host), change the contents of the R2 device by deleting `firstfile` on the target side and replacing it with `secondfile`:

```
‹R2Host#› rm /R2-1/firstfile;touch /R2-1/secondfile
‹R2Host#› ls -l /R2-1

total 0
drwxr-xr-x    2 root     system          256 Jul 20 13:05 lost+found
-rw-r--r--    1 root     system            0 Jul 20 13:55 secondfile

‹R2Host#› umount /R2-1
‹R2Host#› varyoffvg volgrprdf2
‹R2Host#› exportvg volgrprdf2
```

- The `symrdf query` command displays the results of changing the contents of the R2 device.

  There are now local (R1) invalid (modified) tracks on the target (R2) side:

```
‹R2Host#› symrdf -g Rdf2Grp query

Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
```

```
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       4        0 NR 0060 RW       0        0 S...  Split
DEV002  0061 RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 NR 0062 RW       0        0 S...  Split
.
.
.
.
DEV026  0801 RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            4        0                  0        0
  MB(s)             0.2      0.0                0.0      0.0
```

- The `symrdf establish` command performs an incremental establish for the SRDF pairs in device group `Rdf2Grp`, overwriting all changes made to the R2 while the devices were split.

  You can initiate the `establish` action from either the local or remote site, yielding the same results. The individual operations of the `establish` action are logged as they occur.

  For a more detailed report, view the log file in `/var/symapi/log/symapi-yyyymmdd.log`.

‹R2Host#› **symrdf -g Rdf2Grp establish -noprompt**

```
An RDF 'Incremental Establish' operation execution is
in progress for device group 'Rdf2Grp'. Please wait...

    Write Disable device(s) on RA at target (R2)..............Done.
    Suspend RDF link(s)......................................Done.
    Mark target (R2) devices to refresh from source (R1)......Started.
    Devices: 0060-0067, 07E0-07EF in (0256,128)............... Marked.
    Devices: 0800-0803 in (0256,128)......................... Marked.
    Mark target (R2) devices to refresh from source (R1)......Done.
    Merge device track tables between source and target.......Started.
    Devices: 0060-0060 in (0256,128)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Merge device track tables between source and target.......Started.
    Devices: 0061-0067, 07E0-07EF in (0256,128)............... Merged.
    Devices: 0800-0803 in (0256,128)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Done.

The RDF 'Incremental Establish' operation successfully initiated for
device group 'Rdf2Grp'.
```

- The `symrdf verify -summary` command confirms the SRDF pairs are completely synchronized.

‹R2Host#› **symrdf -g Rdf2Grp verify -summary -i 10**

```
Device Group (DG) Name : Rdf2Grp

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     27
  SyncInProg                        1
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            28

  RDF Mode                      Count
  --------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                     28
  -----------------------      ------
  Total                            28

                                 Track(s)    MB(s)
                               -----------  -------
  Total Source R1 Invalid                0      0.0
  Total Source R2 Invalid                4      0.2

  Total Target R1 Invalid                0      0.0
  Total Target R2 Invalid                4      0.2

Not All devices in the group 'Rdf2Grp' are in 'Synchronized' state.

Device Group (DG) Name : Rdf2Grp

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     28
  SyncInProg                        0
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            28


  RDF Mode                      Count
  --------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                     28
  -----------------------      ------
```

```
  Total                              28

Track(s)    MB(s)
                                 -----------  -------
  Total Source R1 Invalid                 0      0.0
  Total Source R2 Invalid                 0      0.0

  Total Target R1 Invalid                 0      0.0
  Total Target R2 Invalid                 0      0.0
```

All devices in the group 'Rdf2Grp' are in 'Synchronized' state.

- The `symrdf split` command and `-ls` commands perform the split and confirm that `secondfile` on the R2 device was removed, and `firstfile` was restored while re-establishing the SRDF device pair:

```
‹R2Host#› symrdf -g Rdf2Grp split -noprompt

An RDF 'Split' operation execution is in progress for device group 'Rdf2Grp'.
Please wait...

    Suspend RDF link(s).......................................Done.
    Read/Write Enable device(s) on RA at target (R2)..........Done.

The RDF 'Split' operation successfully executed for device group 'Rdf2Grp'.

‹R2Host#› importvg -y volgrprdf2 hdisk148
‹R2Host#› mount /dev/fslv00 /R2-1
‹R2Host#› ls -l /R2-1

total 0
-rw-r--r--    1 root     system            0 Jul 20 13:06 firstfile
drwxr-xr-x    2 root     system          256 Jul 20 13:05 lost+found
```

- The following commands replace `firstfile` on the R2 device with `thirdfile` in preparation for demonstrating a `restore` operation:

```
‹R2Host#› rm /R2-1/firstfile;touch /R2-1/thirdfile
‹R2Host#› umount /R2-1
‹R2Host#› varyoffvg volgrprdf2
‹R2Host#› exportvg volgrprdf2
```

- At the local host, the following query displays the results of changing the contents of the R2 device.

  There are now local (R1) invalid tracks on the target (R2) side.

```
‹R1Host#› symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 RW       4        0 S...  Split
```

```
DEV002  0061 RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 NR 0062 RW       0        0 S...  Split
DEV004  0063 RW       0        0 NR 0063 RW       0        0 S...  Split
.
.
.
.
DEV025  0800 RW       0        0 NR 0800 RW       0        0 S...  Split
DEV026  0801 RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            0        0                  4        0
  MB(s)             0.0      0.0                0.2      0.0
```

- The `symrdf restore` command performs an incremental restore, copying tracks that changed on R2 to R1.

  Any tracks on the R1 side that changed while the SRDF pairs were split are overwritten with data from corresponding tracks on the R2 side.

  When the `restore` is complete, R1 contains the same data as R2.

```
‹R1Host#› symrdf -g Rdf1Grp restore -noprompt

An RDF 'Incremental Restore' operation execution is
in progress for device group 'Rdf1Grp'. Please wait...

    Write Disable device(s) on SA at source (R1)..............Done.
    Write Disable device(s) on RA at target (R2)..............Done.
    Suspend RDF link(s)......................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0060-0067, 07E0-07EF in (0321,128)............... Merged.
    Devices: 0800-0803 in (0321,128)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Resume RDF link(s)......................................Done.
    Read/Write Enable device(s) on SA at source (R1)..........Done.

The RDF 'Incremental Restore' operation successfully initiated for
device group 'Rdf1Grp'.
```

- The following commands help to confirm the `restore` operation copied `thirdfile` from the R2 device to the R1 device:

```
‹R1Host#› importvg -y volgrprdf1 hdisk27
‹R1Host#› mount /dev/fslv00 /R1-1
‹R1Host#› ls -l /R1-1

total 0
drwxr-xr-x    2 root     system         256 Jul 20 13:05 lost+found
-rw-r--r--    1 root     system           0 Jul 20 14:02 thirdfile
```

- The `symrdf query` command shows the SRDF pairs are now in the Synchronized state.

  After a `restore` operation, SRDF always places the SRDF pairs in a Synchronized state unlike TimeFinder that places them in the Restored state.

```
‹R1Host#› symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
```

```
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following `failover` action on the target (R2) site write disables the R1 devices in order to move data I/O to the target side:

```
‹R2Host#› symrdf -g Rdf2Grp failover -noprmopt

An RDF 'Failover' operation execution is
in progress for device group 'Rdf2Grp'. Please wait...

    Write Disable device(s) on SA at source (R1)..............Done.
    Suspend RDF link(s)......................................Done.
    Read/Write Enable device(s) on RA at target (R2)..........Done.

The RDF 'Failover' operation successfully executed for
device group 'Rdf2Grp'.
```

- The following query shows the results of the failover: the R1 devices are write disabled (WD), the SRDF links are not ready (NR), and the R2 devices are read write (RW):

```
‹R2Host#› symrdf -g Rdf2Grp query

Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 WD       0        0 S...  Failed Over
DEV002  0061 RW       0        0 NR 0061 WD       0        0 S...  Failed Over
DEV003  0062 RW       0        0 NR 0062 WD       0        0 S...  Failed Over
.
```

```
.
.
.
DEV026  0801 RW       0        0 NR 0801 WD       0        0 S...  Failed Over
DEV027  0802 RW       0        0 NR 0802 WD       0        0 S...  Failed Over
DEV028  0803 RW       0        0 NR 0803 WD       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

While data processing by the remote host continues on the target (R2) side, the system keeps a record of the tracks that changed on the R2 since the failover. The remote (R1) invalid tracks on the target (R2) side are tracks that must be copied from the R2 device to the R1 device after the SRDF links are re-established and a failback is performed. In this example:

- The following interactive script continually runs to rewrite data to the R2 devices.

  ```
  <R2Host#>while true
  > do
  > dd if=/dev/rhdisk148 of=/dev/rhdisk148 bs=1024k count=512
  > done
  ```

- The following query shows the continuous accumulation of remote (R1) invalid tracks (8192) on the target (R2) side:

```
<R2Host#> symrdf -g Rdf2Grp query

Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 NR 0060 WD       0        0 S...  Failed Over
DEV002  0061 RW       0        0 NR 0061 WD       0        0 S...  Failed Over
DEV003  0062 RW       0        0 NR 0062 WD       0        0 S...  Failed Over
.
.
.
.
DEV026  0801 RW       0        0 NR 0801 WD       0        0 S...  Failed Over
DEV027  0802 RW       0        0 NR 0802 WD       0        0 S...  Failed Over
DEV028  0803 RW       0        0 NR 0803 WD       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)         8192        0                  0        0
  MB(s)           512.0      0.0                0.0      0.0
```

- While the R2 side remains accessible for reads and writes, the following `symrdf update` command takes a one-time snapshot of the remote (R1) invalid tracks on the target (R2) side for each device in the group and copies them to the R1 side.

- The `update` operation minimizes downtime when issuing a `failback` command, which write disables the R2.

```
‹R2Host#› symrdf -g Rdf2Grp update -noprompt

An RDF 'Update R1' operation execution is in progress for device group 'Rdf2Grp'.
   Please wait...

    Suspend RDF link(s).....................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0060-0067, 07E0-07EF in (0256,128)............... Merged.
    Devices: 0800-0803 in (0256,128)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Resume RDF link(s)......................................Done.

The RDF 'Update R1' operation successfully initiated for
device group 'Rdf2Grp'.
```

- The following query shows the beginning of the update session with 8192 invalid tracks on the Source (R1) side that need updating:

```
‹R2Host#› symrdf -g Rdf2Grp query -i 10

Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 RW 0060 WD    8192        0 S...  R1 UpdInProg
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  R1 Updated
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  R1 Updated
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  R1 Updated
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  R1 Updated

Total          -------- --------           -------- --------
  Track(s)         8192        0               8192        0
  MB(s)           512.0      0.0              512.0      0.0
```

- As the update progresses, the number of local invalid tracks on the source (R1) side decreases as the tracks are counted down from the initial snapshot.

```
Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
```

```
       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 RW 0060 WD     567        0 S...  R1 UpdInProg
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  R1 Updated
DEV004  0063 RW       0        0 RW 0063 WD       0        0 S...  R1 Updated
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  R1 Updated
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  R1 Updated
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  R1 Updated

Total          -------- --------           -------- --------
  Track(s)         8192        0               567        0
  MB(s)           512.0      0.0              35.4      0.0


Synchronization rate               :     43.3 MB/S
Estimated time to completion       : 00:00:01
```

- Once the initial 8192 tracks are updated, the local (R1) invalid tracks reach zero on the source (R1) side, signifying the end of the update. During this time, any newly-written tracks on the R2 side continue being marked as remote (R1) invalid tracks on the target (R2) side.

```
Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)

Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
```

```
       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 RW 0060 WD       0        0 S...  R1 Updated
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  R1 Updated
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  R1 Updated
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  R1 Updated
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  R1 Updated
```

```
Total          -------- --------           -------- --------
  Track(s)         8192        0                  0        0
  MB(s)           512.0      0.0                0.0      0.0
```

To demonstrate the `update -until` option, this example keeps running continuous I/O to the R2 devices and employs two displays:

- A query showing the update cycle progress, and
- The update showing the continuing output from the `symrdf update -until` command.

  In the following example, the query shows the initial status of the SRDF pairs.

  As updates occur, this information changes and is shown in intervals of 10 seconds.

  - Remote (R1) invalid tracks on the target (R2) side represent continuous I/O to the R2 devices.
  - Local (R1) invalid tracks on the source (R1) side represent the number of tracks that still need to be copied from the target (R2) side (currently zero until the update begins).

‹R2Host#› **symrdf -g Rdf2Grp query -i 10**

```
Device Group (DG) Name             : Rdf2Grp
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600321    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    3444        0 RW 0060 WD       0        0 S...  R1 Updated
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV018  07E9 RW       0        0 RW 07E9 WD       0        0 S...  R1 Updated
DEV019  07EA RW       0        0 RW 07EA WD       0        0 S...  R1 Updated
```

- The following update display illustrates the `symrdf update` command with the `-until` option track threshold setting of 100 tracks.

  While the target (R2) side remains accessible for reads and writes, the system takes a one-time snapshot of the invalid tracks for each device in the group on the target (R2) side and requests SRDF to begin copying those tracks to the source (R1) side.

  If SRDF finishes fully copying the snapshot of updates to the R1 side and there are still 100 or more R1 invalid tracks on the target (R2) side, the system takes another snapshot and requests SRDF to begin copying another batch of tracks to the R1 side.

‹R2Host#› **symrdf -g Rdf2Grp update -noprompt -until 100**

```
An RDF 'Update R1' operation execution is
in progress for device group 'Rdf2Grp'. Please wait...

    Suspend RDF link(s)......................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0060-0067, 07E0-07EF in (0256,128)............... Merged.
```

```
    Devices: 0800-0803 in (0256,128)........................ Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Resume RDF link(s)......................................Done.

The RDF 'Update R1' operation successfully initiated for
device group 'Rdf2Grp'.
```

- The following query indicates the first batch of updates were fully copied to the R1 side.

  The local (R1) invalid track count on the R1 side is zero. Because of continuous I/O on the R2 side during the update cycle, the R1 invalid track count is 3444 and still over the 100-track threshold. Therefore, the system automatically initiates another update cycle, as shown in the next display.

```
Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    3444        0 RW 0060 WD       0        0 S...  R1 Updated
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  R1 Updated
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  R1 Updated
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  R1 Updated
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  R1 Updated

Total          -------- --------           -------- --------
  Track(s)         3444        0                  0        0
  MB(s)           215.2      0.0                0.0      0.0
```

- The following update display indicates the beginning of the second update cycle:

```
An RDF 'Update R1' operation execution is
in progress for device group 'Rdf2Grp'. Please wait...

    Suspend RDF link(s)....................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0060-0067, 07E0-07EF in (0256,128)............. Merged.
    Devices: 0800-0803 in (0256,128)........................ Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Resume RDF link(s)......................................Done.

The RDF 'Update R1' operation successfully initiated for
device group 'Rdf2Grp'.
```

- Because the `update -until` was set at 100, this last batch of 48 updates was not copied to the R1 side.

```
       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------
```

```
DEV001  0060 RW      48        0 RW 0060 WD       0        0 S...  R1 Updated
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  R1 Updated
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  R1 Updated
DEV004  0063 RW       0        0 RW 0063 WD       0        0 S...  R1 Updated
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  R1 Updated
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  R1 Updated
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  R1 Updated

Total          -------- --------           -------- --------
  Track(s)           48        0                  0        0
  MB(s)             3.0      0.0                0.0      0.0
```

Example 2: Concurrent SRDF

The hardware configuration of this concurrent SRDF example consists of the following:

- Local source Symmetrix array (sid 321)
- Remote target Symmetrix array (sid 198)
- Remote target Symmetrix array (sid 256)

All commands are issued from the local site host.

In the following example, the `symcfg list -v` command displays the characteristics of the three Symmetrix arrays in the concurrent SRDF relationship.

Each Symmetrix array in this configuration has the Concurrent RDF Configuration State set to Enabled, a prerequisite for establishing the concurrent SRDF pairs.

```
‹R1Host#› symcfg list -v

Symmetrix ID: 000192600321 (Local)
Time Zone   : CDT

    Product Model                       : VMAX-1
    Symmetrix ID                        : 000192600321

    Microcode Version (Number)          : 5875 (16F30000)
    Microcode Registered Build          : 0
    Microcode Date                      : 07.23.2010

    Microcode Patch Date                : 07.23.2010
    Microcode Patch Level               : 50

    Symmwin Version                     : 25
    Service Processor Time Offset       : - 12214 days 09:52:36

    Cache Size (Mirrored)               :   24576 (MB)
    # of Available Cache Slots          :  315536
    Max # of System Write Pending Slots :  189814
    Max # of DA Write Pending Slots     :       0
    Max # of Device Write Pending Slots :    9490

    Symmetrix Total Operating Time      : 9 days, 03:47:26
    Symmetrix Power ON Time             : Wed Jul 14 08:22:04 2010
    Symmetrix Last IPL Time (Cold)      : Wed Jul 14 08:36:43 2010
    Symmetrix Last Fast IPL Time (Hot)  : Sun Dec 31 18:00:00 2006
```

```
    Host DB Sync Time                     : Fri Jul 23 12:09:26 2010
    Symmetrix CLI (SYMCLI) Version        : X7.2.0.378 (Edit Level: 1101)
    Built with SYMAPI Version             : X7.2.0.378 (Edit Level: 1101)
    SYMAPI Run Time Version               : X7.2.0.378 (Edit Level: 1101)
    SYMAPI Server Version                 : unavailable

    Number of Configured (Sym) Devices    :  2953
    Number of Visible (Host) Devices      :   145
    Number of Configured Actual Disks     :    80
    Number of Configured Hot Spares       :    10
    Number of Unconfigured Disks          :     0
    Maximum number of hypers per disk     :   512
    Front door LED status                 : N/A

    Number of Powerpath Devices           :     0
    Powerpath Run Time Version            : N/A

    SDDF Configuration State              : Enabled
    Configuration Change State            : Enabled
    WORM Configuration Level              : N/A
    WORM Characteristics                  : N/A

    Symmetrix Configuration Checksum      : B0AD26A4
    Switched RDF Configuration State      : Enabled
    Concurrent RDF Configuration State    : Enabled
    Dynamic RDF Configuration State       : Enabled
    Concurrent Dynamic RDF Configuration : Enabled
    RDF Data Mobility Configuration State: Disabled
    Access Control Configuration State    : Enabled
    Device Masking (ACLX) Config State    : Mixed
    Multi LRU Device Assignment           : NONE
    Disk Group Assignments                : In Use
    Hot Swap Policy for Hard Drives       : Permanent
    Hot Swap Policy for Flash Drives      : Permanent
    Symmetrix Disk Library                : Disabled
    FBA Geometry Emulation                : Native
    3 Dynamic Mirrors                     : Enabled
    Cache Partitioning                    : Disabled
    IPSec Status                          : Pass Thru
    Allow spare in mirror 4 position      : Disabled
    Disks Service                         : Normal
    Symmetrix Data Encryption             : Disabled
    Power Save Mode                       : Disabled
    Power Save Idle Time                  : 00:30
    Power Save Unused Drives              : Enabled

   Parity Raid Configuration             : N/A
    Raid-5 Configuration                  : RAID-5 (7+1)
    Raid-6 Configuration                  : RAID-6 (6+2)
    PAV Mode                              : DynamicStandardPAV
    PAV Alias Limit                       : 31

    SRDF/A Maximum Host Throttle (Secs)   : 0
    SRDF/A Maximum Cache Usage (Percent)  : 75

    Auto Meta                             : Disabled
    Minimum Auto Meta Size                : 262669
    Auto Meta Member Size                 : 0
    Auto Meta Configuration               : N/A


Symmetrix ID: 000192600198 (Remote)
Time Zone   : CDT

    Product Model                         : VMAX-1
    Symmetrix ID                          : 000192600198
```

```
Microcode Version (Number)           : 5875 (16F30000)
Microcode Registered Build           : 0
Microcode Date                       : 07.23.2010

Microcode Patch Date                 : 07.23.2010
Microcode Patch Level                : 50

Symmwin Version                      : 25
Service Processor Time Offset        : - 12214 days 20:08:22

Cache Size (Mirrored)                :   24576 (MB)
# of Available Cache Slots           :  319384
Max # of System Write Pending Slots  :  192129
Max # of DA Write Pending Slots      :       0
Max # of Device Write Pending Slots  :    9605

Symmetrix Total Operating Time       : 9 days, 04:12:22
Symmetrix Power ON Time              : Wed Jul 14 07:57:08 2010
Symmetrix Last IPL Time (Cold)       : Wed Jul 14 08:22:07 2010
Symmetrix Last Fast IPL Time (Hot)   : Sun Dec 31 18:00:00 2006

Host DB Sync Time                    : Fri Jul 23 12:09:26 2010
Symmetrix CLI (SYMCLI) Version       : X7.2.0.378 (Edit Level: 1101)
Built with SYMAPI Version            : X7.2.0.378 (Edit Level: 1101)
SYMAPI Run Time Version              : X7.2.0.378 (Edit Level: 1101)
SYMAPI Server Version                : unavailable

Number of Configured (Sym) Devices   :  2953
Number of Visible (Host) Devices     :     0
Number of Configured Actual Disks    :    80
Number of Configured Hot Spares      :    10
Number of Unconfigured Disks         :     0
Maximum number of hypers per disk    :   512
Front door LED status                : N/A

Number of Powerpath Devices          :     0
Powerpath Run Time Version           : N/A

SDDF Configuration State             : Enabled
Configuration Change State           : Enabled
WORM Configuration Level             : N/A
WORM Characteristics                 : N/A

Symmetrix Configuration Checksum     : 55E793F0
Switched RDF Configuration State     : Enabled
Concurrent RDF Configuration State   : Enabled
Dynamic RDF Configuration State      : Enabled
Concurrent Dynamic RDF Configuration : Enabled
RDF Data Mobility Configuration State: Disabled
Access Control Configuration State   : Enabled
Device Masking (ACLX) Config State   : Mixed
Multi LRU Device Assignment          : NONE
Disk Group Assignments               : In Use
Hot Swap Policy for Hard Drives      : Permanent
Hot Swap Policy for Flash Drives     : Permanent
Symmetrix Disk Library               : Disabled
FBA Geometry Emulation               : Native
3 Dynamic Mirrors                    : Enabled
Cache Partitioning                   : Disabled
IPSec Status                         : Pass Thru
Allow spare in mirror 4 position     : Disabled
Disks Service                        : Normal
Symmetrix Data Encryption            : Disabled
Power Save Mode                      : Disabled
Power Save Idle Time                 : 00:30
```

```
    Power Save Unused Drives             : Enabled

    Parity Raid Configuration            : N/A
    Raid-5 Configuration                 : RAID-5 (7+1)
    Raid-6 Configuration                 : RAID-6 (6+2)
    PAV Mode                             : DynamicStandardPAV
    PAV Alias Limit                      : 31

    SRDF/A Maximum Host Throttle (Secs)  : 0
    SRDF/A Maximum Cache Usage (Percent) : 75

    Auto Meta                            : Disabled
    Minimum Auto Meta Size               : 262669
    Auto Meta Member Size                : 0
    Auto Meta Configuration              : N/A

Symmetrix ID: 000192600256 (Remote)
Time Zone   : CDT

    Product Model                        : VMAX-1
    Symmetrix ID                         : 000192600256

    Microcode Version (Number)           : 5875 (16F30000)
    Microcode Registered Build           : 0
    Microcode Date                       : 07.23.2010

    Microcode Patch Date                 : 07.23.2010
    Microcode Patch Level                : 50

    Symmwin Version                      : 25
    Service Processor Time Offset        : - 12214 days 08:53:30

    Cache Size (Mirrored)                :   24576 (MB)
    # of Available Cache Slots           :  315536
    Max # of System Write Pending Slots  :  189814
    Max # of DA Write Pending Slots      :       0
    Max # of Device Write Pending Slots  :    9490

    Symmetrix Total Operating Time       : 9 days, 03:21:20
    Symmetrix Power ON Time              : Wed Jul 14 08:48:10 2010
    Symmetrix Last IPL Time (Cold)       : Wed Jul 14 09:02:06 2010
    Symmetrix Last Fast IPL Time (Hot)   : Sun Dec 31 18:00:00 2006

    Host DB Sync Time                    : Fri Jul 23 12:09:27 2010
    Symmetrix CLI (SYMCLI) Version       : X7.2.0.378 (Edit Level: 1101)
    Built with SYMAPI Version            : X7.2.0.378 (Edit Level: 1101)
    SYMAPI Run Time Version              : X7.2.0.378 (Edit Level: 1101)
    SYMAPI Server Version                : unavailable

    Number of Configured (Sym) Devices   :  2953
    Number of Visible (Host) Devices     :     0
    Number of Configured Actual Disks    :    80
    Number of Configured Hot Spares      :    10
    Number of Unconfigured Disks         :     0
    Maximum number of hypers per disk    :   512
    Front door LED status                : N/A

    Number of Powerpath Devices          :     0
    Powerpath Run Time Version           : N/A

    SDDF Configuration State             : Enabled
    Configuration Change State           : Enabled
    WORM Configuration Level             : N/A
    WORM Characteristics                 : N/A

    Symmetrix Configuration Checksum     : 9586EAB4
```

```
    Switched RDF Configuration State     : Enabled
    Concurrent RDF Configuration State   : Enabled
    Dynamic RDF Configuration State      : Enabled
    Concurrent Dynamic RDF Configuration : Enabled
    RDF Data Mobility Configuration State: Disabled
    Access Control Configuration State   : Enabled
    Device Masking (ACLX) Config State   : Mixed
    Multi LRU Device Assignment          : NONE
    Disk Group Assignments               : In Use
    Hot Swap Policy for Hard Drives      : Permanent
    Hot Swap Policy for Flash Drives     : Permanent
    Symmetrix Disk Library               : Disabled
    FBA Geometry Emulation               : Native
    3 Dynamic Mirrors                    : Enabled
    Cache Partitioning                   : Disabled
    IPSec Status                         : Pass Thru
    Allow spare in mirror 4 position     : Disabled
    Disks Service                        : Normal
    Symmetrix Data Encryption            : Disabled
    Power Save Mode                      : Disabled
    Power Save Idle Time                 : 00:30
    Power Save Unused Drives             : Disabled

Parity Raid Configuration            : N/A
    Raid-5 Configuration                 : RAID-5 (7+1)
    Raid-6 Configuration                 : RAID-6 (6+2)
    PAV Mode                             : DynamicStandardPAV
    PAV Alias Limit                      : 31

    SRDF/A Maximum Host Throttle (Secs)  : 0
    SRDF/A Maximum Cache Usage (Percent) : 75

    Auto Meta                            : Disabled
    Minimum Auto Meta Size               : 262669
    Auto Meta Member Size                : 0
    Auto Meta Configuration              : N/A
```

- The `symrdf list` command with the `-concurrent` option shows the concurrent SRDF devices on the local Symmetrix array.

  Each device of a concurrent pair belongs to a different RDF group, as shown in the RDF Typ:G column.

```
<R1Host#> symrdf list -concurrent -sid 321

Symmetrix ID: 000192600321

                            Local Device View
---------------------------------------------------------------------------
                     STATUS     MODES                        RDF  S T A T E S
Sym        RDF       ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G     SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------   ---------  ----- -------  ------- --- ---- -------------

0060 0060   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0060   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0061 0061   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0061   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0062 0062   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0062   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0063 0063   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0063   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
.
.
.
.
0800 0800   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0800   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0801 0801   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0801   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0802 0802   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0802   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized
0803 0803   R1:128   RW RW RW   S..1.       0        0 RW  WD   Synchronized
     0803   R1:228   RW RW RW   S..1.       0        0 RW  WD   Synchronized

Total                                 -------- --------
  Track(s)                                   0        0
  MB(s)                                    0.0      0.0
```

- The `sympd list` command displays all devices visible to the local host.

  The N/Grp'd attribute indicates that these devices are not associated with any device group and can be added to one:

```
‹R1Host#› sympd list -sid 321

Symmetrix ID: 000192600321

        Device Name           Directors                  Device
--------------------------- ------------- ------------------------------------
                                                                        Cap
Physical                Sym  SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- ------------------------------------

/dev/rhdisk27           0060 10E:0 10A:D0  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk28           0061 10E:0 09C:C0  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk29           0062 10E:0 10B:C0  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk30           0063 10E:0 09D:D0  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk31           0064 10E:0 09A:C2  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk32           0065 10E:0 10C:D2  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk33           0066 10E:0 09B:D2  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk34           0067 10E:0 10D:C2  RDF1+Mir      N/Grp'd      RW    4314
/dev/rhdisk43           07E0 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
```

```
/dev/rhdisk44          07E1 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk45          07E2 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk46          07E3 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk47          07E4 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk48          07E5 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk49          07E6 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk50          07E7 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk51          07E8 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk52          07E9 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk53          07EA 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk54          07EB 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk55          07EC 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk56          07ED 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk57          07EE 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk58          07EF 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk76          0800 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk216         0801 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk217         0802 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
/dev/rhdisk218         0803 10E:0  NA:NA  RDF1+TDEV     N/Grp'd      RW    4314
```

Creating a device group and adding devices to it are prerequisites for performing SRDF and TimeFinder operations.

- The `symdg create` command creates a device group named `conrdf`, and the `symdg addall` command adds all devices belonging to RDF group 128 to `conrdf`:

  ```
  <R1Host#> symdg create conrdf -type rdf1
  <R1Host#> symdg -g conrdf -sid 321 addall dev -sel_rdfg 128
  ```

- The following query displays the status of the SRDF pairs in the `conrdf` device group.

  The `-rdfg all` option ensures the display shows the status of both links of the concurrent SRDF pair.

```
<R1Host#> symrdf -g conrdf query -rdfg all

Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 228 (E3)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
             RW       0        0 RW 0060 WD       0        0 S...  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
             RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
             RW       0        0 RW 0062 WD       0        0 S...  Synchronized
DEV004  0063 RW       0        0 RW 0063 WD       0        0 S...  Synchronized
.
.
.
.

DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
```

```
             RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
             RW       0        0 RW 0802 WD       0        0 S...  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized
             RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following `symrdf split` command splits the SRDF pairs in the `conrdf` device group. You can split a concurrent SRDF pair either simultaneously or sequentially. The `-rdfg all` option causes both concurrent devices of the SRDF concurrent pair to be split at the same time.

```
‹R1Host#› symrdf -g conrdf split -rdfg all -noprompt

An RDF 'Split' operation execution is
in progress for device group 'conrdf'. Please wait...

    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Read/Write Enable device(s) in (0321,128) on RA at target (R2)...Done.
    Read/Write Enable device(s) in (0321,228) on RA at target (R2)...Done.

The RDF 'Split' operation successfully executed for
device group 'conrdf'.
```

- The following `symrdf establish` command performs a full establish on the SRDF pairs in the `conrdf` device group. You can establish a concurrent SRDF pair either simultaneously or sequentially. The `-rdfg all` option causes both concurrent devices of the SRDF concurrent pair to be established simultaneously.

```
‹R1Host#› symrdf -g conrdf establish -full -rdfg all -noprompt

An RDF 'Full Establish' operation execution is
in progress for device group 'conrdf'. Please wait...

    Write Disable device(s) in (0321,128) on RA at target (R2).......Done.
    Write Disable device(s) in (0321,228) on RA at target (R2).......Done.
    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Suspend RDF link(s) for device(s) in (0321,228)..................Done.
    Resume RDF link(s) for device(s) in (0321,128)...................Started.
    Resume RDF link(s) for device(s) in (0321,228)...................Started.
    Merge track tables between source and target in (0321,228).......Started.
    Merge track tables between source and target in (0321,128).......Started.
    Devices: 0060-0067, 07E0-07EF in (0321,228)..................... Merged.
    Devices: 0800-0803 in (0321,228)................................ Merged.
    Devices: 0060-0067, 07E0-07EF in (0321,128)..................... Merged.
    Devices: 0800-0803 in (0321,128)................................ Merged.
    Merge track tables between source and target in (0321,228).......Done.
    Merge track tables between source and target in (0321,128).......Done.
    Resume RDF link(s) for device(s) in (0321,228)...................Done.
    Resume RDF link(s) for device(s) in (0321,128)...................Done.

The RDF 'Full Establish' operation successfully initiated for
device group 'conrdf'.
```

- The following query shows the concurrent SRDF pairs are in the process of synchronizing (SyncInProg):

```
‹R1Host#› symrdf -g conrdf query -rdfg all

Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
```

```
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 228 (E3)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0    69030 RW 0060 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0060 WD       0        0 S...  SyncInProg
DEV002  0061 RW       0    69030 RW 0061 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0061 WD       0        0 S...  SyncInProg
DEV003  0062 RW       0    69030 RW 0062 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0062 WD       0        0 S...  SyncInProg
.
.
.
.
DEV021  07EC RW       0    69030 RW 07EC WD       0        0 S...  SyncInProg
             RW       0    69030 RW 07EC WD       0        0 S...  SyncInProg
DEV022  07ED RW       0    68863 RW 07ED WD       0        0 S...  SyncInProg
             RW       0    68863 RW 07ED WD       0        0 S...  SyncInProg
DEV023  07EE RW       0    69030 RW 07EE WD       0        0 S...  SyncInProg
             RW       0    69030 RW 07EE WD       0        0 S...  SyncInProg
DEV024  07EF RW       0    68671 RW 07EF WD       0        0 S...  SyncInProg
             RW       0    68671 RW 07EF WD       0        0 S...  SyncInProg
DEV025  0800 RW       0    69030 RW 0800 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0800 WD       0        0 S...  SyncInProg
DEV026  0801 RW       0    69030 RW 0801 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0801 WD       0        0 S...  SyncInProg
DEV027  0802 RW       0    69030 RW 0802 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0802 WD       0        0 S...  SyncInProg
DEV028  0803 RW       0    69030 RW 0803 WD       0        0 S...  SyncInProg
             RW       0    69030 RW 0803 WD       0        0 S...  SyncInProg

Total          -------- --------           -------- --------
  Track(s)            0  3861954                  0        0
  MB(s)               0   241372                  0        0
```

- The following `symrdf verify -summary` command displays a summary message every 30 seconds until both concurrent mirrors of each SRDF pair are synchronized:

```
‹R1Host#› symrdf -g conrdf verify -summary -rdfg all -i 30 -synchronized

Device Group (DG) Name : conrdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                      0
  SyncInProg                       56
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
```

```
  -----------------------      ------
  Total                            56

  RDF Mode                      Count
  ---------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                     56
  -----------------------      ------
  Total                            56

                                 Track(s)    MB(s)
                               -----------  -------

  Total Source R1 Invalid                0        0
  Total Source R2 Invalid          3794765   237173

  Total Target R1 Invalid                0        0
  Total Target R2 Invalid                0        0

None of the devices in the group 'conrdf' are in 'Synchronized' state.

Device Group (DG) Name : conrdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                      0
  SyncInProg                       56
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            56

  RDF Mode                      Count
  ---------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                     56
  -----------------------      ------
  Total                            56

                                 Track(s)    MB(s)
                               -----------  -------

Total Source R1 Invalid                0        0
  Total Source R2 Invalid          3748461   234279

  Total Target R1 Invalid                0        0
  Total Target R2 Invalid                0        0

  Synchronization rate         :     90.4 MB/S
  Estimated time to completion : 00:43:11

None of the devices in the group 'conrdf' are in 'Synchronized' state.
```

```
Device Group (DG) Name : conrdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     10
  SyncInProg                       46
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            56

  RDF Mode                      Count
  --------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                     56
  -----------------------      ------
  Total                            56

                                 Track(s)    MB(s)
                               -----------  -------

Total Source R1 Invalid                0        0
  Total Source R2 Invalid          1811934   113246

  Total Target R1 Invalid                0        0
  Total Target R2 Invalid                0        0

  Synchronization rate         :    262.5 MB/S
  Estimated time to completion : 00:09:31

Not All devices in the group 'conrdf' are in 'Synchronized' state.

Device Group (DG) Name : conrdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     55
  SyncInProg                        1
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            56

  RDF Mode                      Count
  --------------------------  ------
  Adaptive Copy Disk                0
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
```

```
   Synchronous                        56
  -----------------------      ------
  Total                            56

                                 Track(s)    MB(s)
                               -----------  -------

  Total Source R1 Invalid                0      0.0
  Total Source R2 Invalid              272     17.0

  Total Target R1 Invalid                0      0.0
  Total Target R2 Invalid                0      0.0

Not All devices in the group 'conrdf' are in 'Synchronized' state.

Device Group (DG) Name : conrdf

  RDF Pair State                 Count
  -----------------------      ------
  Consistent                         0
  Synchronized                      56
  SyncInProg                         0
  Suspended                          0
  Split                              0
  Failed Over                        0
  R1 Updated                         0
  R1 UpdInProg                       0
  TransIdle                          0
  Partitioned                        0
  Invalid                            0
  -----------------------      ------
  Total                             56


  RDF Mode                       Count
  ---------------------------  ------
  Adaptive Copy Disk                 0
  Adaptive Copy Write Pending        0
  Asynchronous                       0
  Semi-Synchronous                   0
   Synchronous                      56
  -----------------------      ------
  Total                             56

                                 Track(s)    MB(s)
                               -----------  -------
  Total Source R1 Invalid                0      0.0
  Total Source R2 Invalid                0      0.0

  Total Target R1 Invalid                0      0.0
  Total Target R2 Invalid                0      0.0
```

All devices in the group 'conrdf' are in 'Synchronized' state.

- The following query confirms that both concurrent SRDF pairs are in the Synchronized state:

‹R1Host#› **symrdf -g conrdf query -rdfg all**

```
Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 228 (E3)
```

```
       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
             RW       0        0 RW 0060 WD       0        0 S...  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
             RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
             RW       0        0 RW 0062 WD       0        0 S...  Synchronized
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
             RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
             RW       0        0 RW 0802 WD       0        0 S...  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized
             RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf split` command splits the SRDF pairs in the `conrdf` device group. The `-rdfg all` option causes both concurrent devices of the SRDF concurrent pair to be split simultaneously.

```
‹R1Host#› symrdf -g conrdf split -rdfg all -noprompt

An RDF 'Split' operation execution is
in progress for device group 'conrdf'. Please wait...

    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Read/Write Enable device(s) in (0321,128) on RA at target (R2)...Done.
    Read/Write Enable device(s) in (0321,228) on RA at target (R2)...Done.

The RDF 'Split' operation successfully executed for
device group 'conrdf'.
```

- You can restore from one of the concurrent R2 mirrors at any given time. The following `symrdf restore` command with the `-rdfg 128` option causes a restore operation from the concurrent R2 mirror whose RDF group is 128:

```
‹R1Host#› symrdf -g conrdf restore -rdfg 128 -noprompt

An RDF 'Incremental Restore' operation execution is
in progress for device group 'conrdf'. Please wait...

    Write Disable device(s) in (0321,128) on SA at source (R1).......Done.
    Write Disable device(s) in (0321,128) on RA at target (R2).......Done.
```

```
    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Mark source device(s) in (0321,128) to refresh from target.......Started.
    Devices: 0060-0060 in (0321,128)............................... Marked.
    Mark source device(s) in (0321,128) to refresh from target.......Done.
    Merge track tables between source and target in (0321,128).......Started.
    Devices: 0060-0067, 07E0-07EF in (0321,128)..................... Merged.
    Devices: 0800-0803 in (0321,128)............................... Merged.
    Merge track tables between source and target in (0321,128).......Done.
    Resume RDF link(s) for device(s) in (0321,128)...................Started.
    Resume RDF link(s) for device(s) in (0321,128)...................Done.
    Read/Write Enable device(s) in (0321,128) on SA at source (R1)...Done.

The RDF 'Incremental Restore' operation successfully initiated for
device group 'conrdf'.

An RDF 'Incremental Restore' operation execution is in progress for device
   group 'conrdf'. Please wait...

    Write Disable device(s) in (0077,02) on SA at source (R1).......Done.
    Write Disable device(s) in (0077,02) on RA at target (R2).......Done.
    Suspend RDF link(s) for device(s) in (0077,02)..................Done.
    Merge track tables between source and target in (0077,02).......Started.
    Devices: 0028-0029 ........................................... Merged.
    Merge track tables between source and target in (0077,02).......Done.
    Resume RDF link(s) for device(s) in (0077,02)...................Done.
    Read/Write Enable device(s) in (0077,02) on SA at source (R1)...Done.

The RDF 'Incremental Restore' operation successfully initiated for device
   group 'conrdf'.
```

- The following query with the `-rdfg 128` option shows the status of each concurrent R2 mirror whose link is represented by the SRDF group 128:

‹R1Host#› **symrdf -g conrdf query -rdfg 128**

```
Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 RW 0060 WD       0        0 S...  SyncInProg
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
DEV004  0063 RW       0        0 RW 0063 WD       0        0 S...  Synchronized
DEV005  0064 RW       0        0 RW 0064 WD       0        0 S...  Synchronized
DEV006  0065 RW       0        0 RW 0065 WD       0        0 S...  Synchronized
DEV007  0066 RW       0        0 RW 0066 WD       0        0 S...  Synchronized
DEV008  0067 RW       0        0 RW 0067 WD       0        0 S...  Synchronized
DEV009  07E0 RW       0        0 RW 07E0 WD       0        0 S...  Synchronized
DEV010  07E1 RW       0        0 RW 07E1 WD       0        0 S...  Synchronized
DEV011  07E2 RW       0        0 RW 07E2 WD       0        0 S...  Synchronized
DEV012  07E3 RW       0        0 RW 07E3 WD       0        0 S...  Synchronized
DEV013  07E4 RW       0        0 RW 07E4 WD       0        0 S...  Synchronized
DEV014  07E5 RW       0        0 RW 07E5 WD       0        0 S...  Synchronized
DEV015  07E6 RW       0        0 RW 07E6 WD       0        0 S...  Synchronized
DEV016  07E7 RW       0        0 RW 07E7 WD       0        0 S...  Synchronized
```

```
DEV017  07E8 RW       0        0 RW 07E8 WD       0        0 S...  Synchronized
DEV018  07E9 RW       0        0 RW 07E9 WD       0        0 S...  Synchronized
DEV019  07EA RW       0        0 RW 07EA WD       0        0 S...  Synchronized
DEV020  07EB RW       0        0 RW 07EB WD       0        0 S...  Synchronized
DEV021  07EC RW       0        0 RW 07EC WD       0        0 S...  Synchronized
DEV022  07ED RW       0        0 RW 07ED WD       0        0 S...  Synchronized
DEV023  07EE RW       0        0 RW 07EE WD       0        0 S...  Synchronized
DEV024  07EF RW       0        0 RW 07EF WD       0        0 S...  Synchronized
DEV025  0800 RW       0        0 RW 0800 WD       0        0 S...  Synchronized
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)         8192        0                  0        0
  MB(s)           512.0      0.0                0.0      0.0
```

- The following query shows the status of all links of each concurrent SRDF pair. The concurrent R2 mirror from which the restore occurred is now synchronized with its R1 device (Synchronized state). The other concurrent mirrors are still in the Split state:

‹R1Host#› **symrdf -g conrdf query -rdfg all**

```
Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 228 (E3)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                   LI      ST
Standard      A                    N       A
Logical       T  R1 Inv   R2 Inv   K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks   S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
             RW       0     8192 NR 0060 RW       0        0 S...  Split
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
             RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
             RW       0        0 NR 0062 RW       0        0 S...  Split
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
             RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
             RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized
             RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            0     8192                  0        0
  MB(s)             0.0    512.0                0.0      0.0
```

- If you have written new data to the concurrent R2 mirror that is still in the Split state, and you want this data to become resynchronized data, you can restore again from the split mirror. In this case, include the `-remote` option on the `symrdf restore` command line to indicate that you intend to copy data from the split concurrent mirror

to both the R1 device and the other (synchronized) concurrent mirror. The `-rdfg 128` option causes the restore operation to occur now from the concurrent mirror whose link is represented by SRDF group 128.

```
‹R1Host#› symrdf -g conrdf restore -rdfg 128 -noprompt

An RDF 'Incremental Restore' operation execution is
in progress for device group 'conrdf'. Please wait...

    Write Disable device(s) in (0321,128) on SA at source (R1).......Done.
    Write Disable device(s) in (0321,128) on RA at target (R2).......Done.
    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Mark source device(s) in (0321,128) to refresh from target.......Started.
    Devices: 0060-0060 in (0321,128)............................... Marked.
    Mark source device(s) in (0321,128) to refresh from target.......Done.
    Merge track tables between source and target in (0321,128).......Started.
    Devices: 0060-0067, 07E0-07EF in (0321,128)..................... Merged.
    Devices: 0800-0803 in (0321,128)............................... Merged.
    Merge track tables between source and target in (0321,128).......Done.
    Resume RDF link(s) for device(s) in (0321,128)...................Started.
    Resume RDF link(s) for device(s) in (0321,128)...................Done.
    Read/Write Enable device(s) in (0321,128) on SA at source (R1)...Done.

The RDF 'Incremental Restore' operation successfully initiated for
device group 'conrdf'.
    Write Disable device(s) in (0077,01) on SA at source (R1).......Done.
    Write Disable device(s) in (0077,01) on RA at target (R2).......Done.
    Suspend RDF link(s) for device(s) in (0077,01)..................Done.
    Merge track tables between source and target in (0077,01).......Started.
    Devices: 0028-0029 ........................................... Merged.
    Merge track tables between source and target in (0077,01).......Done.
    Resume RDF link(s) for device(s) in (0077,01)...................Done.
    Read/Write Enable device(s) in (0077,01) on SA at source (R1)...Done.

The RDF 'Incremental Restore' operation successfully initiated for device
   group 'conrdf'.
```

- The following `symrdf verify` command with the `-rdfg 128` option displays the RDF Pair State of each concurrent R2 mirror represented by RDF group 128:

```
‹R1Host#› symrdf -g conrdf query -rdfg 128

Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW    8192        0 RW 0060 WD       0        0 S...  SyncInProg
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
.
.
.
.DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
```

```
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)         8192        0                  0        0
  MB(s)           512.0      0.0                0.0      0.0
```

- The following query displays the RDF Pair State of both links of the concurrent SRDF pairs. The links belonging to RDF group 128 are Synchronized whereas the links belonging to SRDF group 228 are Split.

‹R1Host#› **symrdf -g conrdf query -rdfg all**

```
Device Group (DG) Name             : conrdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 228 (E3)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
             RW       0     8192 NR 0060 RW       0        0 S...  Split
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
             RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 RW 0062 WD       0        0 S...  Synchronized
             RW       0        0 NR 0062 RW       0        0 S...  Split
.
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 S...  Synchronized
             RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 RW 0802 WD       0        0 S...  Synchronized
             RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 RW 0803 WD       0        0 S...  Synchronized
             RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            0     8192                  0        0
  MB(s)             0.0    512.0                0.0      0.0
```

- In the context of the device group, you can associate a remote BCV with one of the R2 mirrors of a concurrent SRDF pair, but not with both mirrors. Consequently, your device group can include a BCV that belongs to one of the SRDF groups, but not both. The following `symbcv associate` command includes a remotely-associated (`-rdf`) BCV device in the `conrdf` device group:

‹R1Host#› **symbcv -g conrdf associate dev 250 -rdfg 128 -rdf -v**

```
Device(s) 00250:00250: ADD DEVS BY RANGE Succeeded.
```

- The following `symmir establish` command fully establishes the standard device DEV001 with the remotely-associated BCV. When there are more standard devices in a device group than BCVs, specify which standard device to establish.

```
‹R1Host#› symmir -g conrdf -full establish -rdf DEV001 -noprompt -v

Remote 'Full Establish' operation execution is in progress for device 'DEV001' in
device group 'conrdf'. Please wait...

SELECTING the list of Source devices in the group:

  Device: 0060 [SELECTED]

SELECTING Target devices in the group:

  Device: 0250 [SELECTED]

PAIRING of Standard and BCV devices:

  Devices: 0060(S) - 0250(B) [PAIRED]

STARTING a BCV 'ESTABLISH' operation.

The BCV 'ESTABLISH' operation SUCCEEDED.

Remote 'Full Establish' operation successfully initiated for device 'DEV001'
in group 'conrdf'.
```

- The following `symmir query` command with the `-rdf` option shows that RBCV001 (device 250) is now synchronized as a BCV pair with the DEV001 remote R2 mirror (device 60):

```
‹R1Host#› symmir -g conrdf query -rdf

Device Group (DG) Name: conrdf
DG's Type             : RDF1
DG's Symmetrix ID     : 000192600321
Remote Symmetrix ID   : 000192600256

                    R E M O T E      S Y M M E T R I X


     Standard Device                    BCV Device                  State
------------------------- ------------------------------------- ------------
                    Inv.                                 Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- ------------

DEV001         0060      0 RBCV001              0250 *         0 Synchronized

Total              -------                                -------
  Track(s)              0                                      0
  MB(s)               0.0                                    0.0

Device Group (DG) Name: conrdf
DG's Type             : RDF1
DG's Symmetrix ID     : 000185400077
Remote Symmetrix ID   : 000185400124

                    R E M O T E      S Y M M E T R I X
MB(s)               0.0                                  0.0

Legend:
(*): The paired BCV device is associated with this group.
```

- The following `symbcv disassociate` command disassociates BCV device 250 from the device group. The BCV pair remains synchronized even though it is no longer under the control of the device group.

Note: After you disassociate all remote BCVs on one concurrent SRDF leg, you can now add remote BCVs from the other concurrent SRDF leg.

```
<R1Host#> symbcv -g conrdf disassociate dev 250 -rdf -v

Device(s) 00250:00250: REMOVE DEVS BY RANGE Succeeded.
```

- The following `symbcv associate` command includes a remotely-associated BCV device that belongs to SRDF group 228 (that is, this BCV resides on the other remote Symmetrix array) in the device group:

```
<R1Host#> symbcv -g conrdf associate dev 251 -rdfg 228 -rdf -v

Device(s) 00251:00251: ADD DEVS BY RANGE Succeeded.
```

- The following `symmir establish` command fully establishes the standard device DEV001 with the remotely-associated BCV 251. When there are more standard devices in a device group than BCVs, specify which standard device you want to establish. This BCV is now the only BCV device under the control of the device group.

```
<R1Host#> symmir -g conrdf -full establish -rdf DEV001 -noprompt -v

Remote 'Full Establish' operation execution is in progress for device 'DEV001' in
device group 'conrdf'. Please wait...

SELECTING the list of Source devices in the group:

  Device: 0060 [SELECTED]

SELECTING Target devices in the group:

  Device: 0251 [SELECTED]

PAIRING of Standard and BCV devices:

  Devices: 0060(S) - 0251(B) [PAIRED]

STARTING a BCV 'ESTABLISH' operation.

The BCV 'ESTABLISH' operation SUCCEEDED.

Remote 'Full Establish' operation successfully initiated for device 'DEV001'
in group 'conrdf'.
```

- The following `symmir query` command with the `-rdf` option shows that RBCV001 (device 0251) is now synchronized as a BCV pair with DEV001's other remote R2 mirror (device 0060):

```
<R1Host#> symmir -g conrdf query -rdf

Device Group (DG) Name: conrdf
DG's Type             : RDF1
DG's Symmetrix ID     : 000192600321
Remote Symmetrix ID   : 000192600198


                    R E M O T E     S Y M M E T R I X


     Standard Device                    BCV Device                  State
------------------------- ------------------------------------- ------------
                     Inv.                                  Inv.
```

```
Logical         Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- ------------

DEV001          0060      0 RBCV001              0251 *         0 Synchronized

Total              -------                                -------
  Track(s)               0                                      0
  MB(s)                0.0                                    0.0

Legend:

(*): The paired BCV device is associated with this group.
```

Example 3: Creating a dynamic SRDF group

The commands in this example use a setup of one local Symmetrix arrays (sid 321) connected to two remote Symmetrix arrays (sids 198 and 256).

- The following `symcfg list` command displays the Symmetrix arrays visible to the host:

‹R1Host#› **symcfg list**

```
                              S Y M M E T R I X

                                      Mcode    Cache      Num Phys  Num Symm
    SymmID       Attachment  Model    Version  Size (MB)  Devices   Devices

    000192600321 Local       VMAX-1   5875       24576       145      2953
    000192600198 Remote      VMAX-1   5875       24576         0      2953
    000192600256 Remote      VMAX-1   5875       24576         0      2953
```

- The following `symcfg list -ra all` command displays the SRDF (RA) groups of all Symmetrix arrays connected to each other on the local host and accessible through SRDF links. The `-switched` option shows if the SRDF group **Type** is dynamic or static.

‹R1Host#› **symcfg list -ra all -switched**

```
Symmetrix ID: 000192600321 (Local)

            S Y M M E T R I X   R D F   D I R E C T O R S


           Local                Group                        Remote
    --------------------   ------------------  ---------------------------------
    Ident  Symb RA Grp     Type    Name        SymmID       Ident  Symb  RA Grp
    ------ ---- --------   ------- ----------  ------------ ------ ---- --------

    RF-7F  07F  128 (7F)   Dynamic SG_128      000192600256 RF-7F  07F  128 (7F)
                228 (E3)   Dynamic SG_228      000192600198 RF-7F  07F  228 (E3)
                  2 (01)   Dynamic RDFGRP01    000192600256 RF-8F  08F    2 (01)
                  2 (01)   Dynamic RDFGRP01    000192600256 RF-7F  07F    2 (01)
                128 (7F)   Dynamic SG_128      000192600256 RF-8F  08F  128 (7F)
                 35 (22)   Static  STATIC_22   000192600256 RF-7F  07F   35 (22)
                228 (E3)   Dynamic SG_228      000192600198 RF-8F  08F  228 (E3)

    RF-8F  08F    2 (01)   Dynamic RDFGRP01    000192600256 RF-7F  07F    2 (01)
                128 (7F)   Dynamic SG_128      000192600256 RF-7F  07F  128 (7F)
                  2 (01)   Dynamic RDFGRP01    000192600256 RF-8F  08F    2 (01)
                128 (7F)   Dynamic SG_128      000192600256 RF-8F  08F  128 (7F)
                228 (E3)   Dynamic SG_228      000192600198 RF-7F  07F  228 (E3)
                228 (E3)   Dynamic SG_228      000192600198 RF-8F  08F  228 (E3)

                129 (80)   Dynamic SG_129      000192600256 RF-10F 10F  129 (80)
```

```
                 229 (E4)   Dynamic SG_229      000192600198 RF-9F  09F  229 (E4)
                 129 (80)   Dynamic SG_129      000192600256 RF-9F  09F  129 (80)
                 229 (E4)   Dynamic SG_229      000192600198 RF-10F 10F  229 (E4)

    RF-10F 10F   129 (80)   Dynamic SG_129      000192600256 RF-9F  09F  129 (80)
                 229 (E4)   Dynamic SG_229      000192600198 RF-9F  09F  229 (E4)
                 229 (E4)   Dynamic SG_229      000192600198 RF-10F 10F  229 (E4)
                 129 (80)   Dynamic SG_129      000192600256 RF-10F 10F  129 (80)
                  27 (1A)   Dynamic al-swap27   000192600256 RF-10F 10F   27 (1A)

Symmetrix ID: 000192600198 (Remote)

            S Y M M E T R I X   R D F   D I R E C T O R S


           Local                Group                      Remote
    --------------------  ------------------  ---------------------------------
    Ident  Symb RA Grp     Type    Name        SymmID       Ident  Symb  RA Grp
    ------ ---- --------   ------- ----------  ------------ ------ ---- --------

    RF-7F  07F  178 (B1)   Dynamic SG_178      000192600256 RF-8F  08F  178 (B1)
                  1 (00)   Dynamic RDFGRP00    000192600256 RF-8F  08F    1 (00)
                  1 (00)   Dynamic RDFGRP00    000192600256 RF-7F  07F    1 (00)
                 178 (B1)  Dynamic SG_178      000192600256 RF-7F  07F  178 (B1)
                 228 (E3)  Dynamic SG_228      000192600321 RF-8F  08F  228 (E3)
                 228 (E3)  Dynamic SG_228      000192600321 RF-7F  07F  228 (E3)

                 228 (E3)  Dynamic SG_228      000192600321 RF-8F  08F  228 (E3)
                  1 (00)   Dynamic RDFGRP00    000192600256 RF-8F  08F    1 (00)
                  1 (00)   Dynamic RDFGRP00    000192600256 RF-7F  07F    1 (00)
                 178 (B1)  Dynamic SG_178      000192600256 RF-8F  08F  178 (B1)
                  33 (20)  Static  STATIC_20   000192600256 RF-8F  08F   33 (20)
                 178 (B1)  Dynamic SG_178      000192600256 RF-7F  07F  178 (B1)
                 228 (E3)  Dynamic SG_228      000192600321 RF-7F  07F  228 (E3)

    RF-9F  09F  229 (E4)   Dynamic SG_229      000192600321 RF-10F 10F  229 (E4)
                 229 (E4)  Dynamic SG_229      000192600321 RF-9F  09F  229 (E4)
                 179 (B2)  Dynamic SG_179      000192600256 RF-10F 10F  179 (B2)
                 179 (B2)  Dynamic SG_179      000192600256 RF-9F  09F  179 (B2)

    RF-10F 10F  229 (E4)   Dynamic SG_229      000192600321 RF-10F 10F  229 (E4)
                 229 (E4)  Dynamic SG_229      000192600321 RF-9F  09F  229 (E4)
                 179 (B2)  Dynamic SG_179      000192600256 RF-10F 10F  179 (B2)
                 179 (B2)  Dynamic SG_179      000192600256 RF-9F  09F  179 (B2)

Symmetrix ID: 000192600256 (Remote)

            S Y M M E T R I X   R D F   D I R E C T O R S


           Local                Group                      Remote
    --------------------  ------------------  ---------------------------------
    Ident  Symb RA Grp     Type    Name        SymmID       Ident  Symb  RA Grp
    ------ ---- --------   ------- ----------  ------------ ------ ---- --------

    RF-7F  07F  128 (7F)   Dynamic SG_128      000192600321 RF-7F  07F  128 (7F)
                  2 (01)   Dynamic RDFGRP01    000192600321 RF-8F  08F    2 (01)
                  1 (00)   Dynamic RDFGRP00    000192600198 RF-8F  08F    1 (00)
                  1 (00)   Dynamic RDFGRP00    000192600198 RF-7F  07F    1 (00)
                  2 (01)   Dynamic RDFGRP01    000192600321 RF-7F  07F    2 (01)
                 178 (B1)  Dynamic SG_178      000192600198 RF-8F  08F  178 (B1)
                  35 (22)  Static  STATIC_22   000192600321 RF-7F  07F   35 (22)
                 128 (7F)  Dynamic SG_128      000192600321 RF-8F  08F  128 (7F)
                 178 (B1)  Dynamic SG_178      000192600198 RF-7F  07F  178 (B1)

    RF-8F  08F  178 (B1)   Dynamic SG_178      000192600198 RF-7F  07F  178 (B1)
```

```
                  2 (01)   Dynamic RDFGRP01    000192600321 RF-8F  08F    2 (01)
                  1 (00)   Dynamic RDFGRP00    000192600198 RF-8F  08F    1 (00)
                  1 (00)   Dynamic RDFGRP00    000192600198 RF-7F  07F    1 (00)
                  2 (01)   Dynamic RDFGRP01    000192600321 RF-7F  07F    2 (01)
                178 (B1)   Dynamic SG_178      000192600198 RF-8F  08F  178 (B1)
                128 (7F)   Dynamic SG_128      000192600321 RF-7F  07F  128 (7F)
                128 (7F)   Dynamic SG_128      000192600321 RF-8F  08F  128 (7F)
                 33 (20)   Static  STATIC_20   000192600198 RF-8F  08F   33 (20)

                179 (B2)   Dynamic SG_179      000192600198 RF-9F  09F  179 (B2)
                179 (B2)   Dynamic SG_179      000192600198 RF-10F 10F  179 (B2)
                129 (80)   Dynamic SG_129      000192600321 RF-9F  09F  129 (80)
                129 (80)   Dynamic SG_129      000192600321 RF-10F 10F  129 (80)

    RF-10F 10F  179 (B2)   Dynamic SG_179      000192600198 RF-9F  09F  179 (B2)
                129 (80)   Dynamic SG_129      000192600321 RF-9F  09F  129 (80)
                179 (B2)   Dynamic SG_179      000192600198 RF-10F 10F  179 (B2)
                129 (80)   Dynamic SG_129      000192600321 RF-10F 10F  129 (80)
                 27 (1A)   Dynamic al-swap27   000192600321 RF-10F 10F   27 (1A)
```

- The following `symrdf addgrp` command creates a dynamic SRDF group that represents another SRDF link between Symmetrix 321 and Symmetrix 256. It adds SRDF group 148 on the local Symmetrix 321 and SRDF group 148 on the remote Symmetrix 256. You must specify a group label (in this case, `DYNGRP148`) to use when modifying or deleting this group. Creation of the local and remote SRDF groups includes directors `7f,8f` from both the local and remote Symmetrix arrays. Note the SRDF group number and the director do not have to be the same on the local and remote Symmetrix arrays.

  When creating dynamic SRDF groups between two Symmetrix arrays, it is important to understand the network topology when choosing director endpoints. If using the Fibre Channel protocol, the director endpoints must see each other through the FC fabric to create the dynamic SRDF links. Ensure that the physical connections between the local RA and remote RA are valid and operational. Static and dynamic groups may co-exist on the same RA directors.

```
‹R1Host#› symrdf addgrp -label DYNGRP148 -sid 321 -rdfg 148 -dir 7f,8f \
-remote_sid 256 -remote_rdfg 148 -remote_dir 7f,8f

Successfully Added Dynamic RDF Group 'DYNGRP148' for Symm: 000192600321
```

- The following `symcfg list -ra all` command with the `-switched` option verifies that the SRDF group 148 (`DYNGRP148`) was added to both the local and remote Symmetrix arrays:

```
‹R1Host#› symcfg -sid 321 list -rdfg 148

Symmetrix ID : 000192600321

                S Y M M E T R I X   R D F   G R O U P S

    Local             Remote                   Group                RDFA Info
-------------- --------------------- -------------------------- ---------------
            LL                                   Flags   Dir Flags Cycle
 RA-Grp  (sec)  RA-Grp  SymmID       T    Name    LPDS CHT  Cfg CSRM  time  Pri
-------------- --------------------- -------------------------- ----- ----- ---
148 (93)    10 148 (93) 000192600256 D DYNGRP148  XX.. ..X  F-S -IS-     30  33

Legend:
  ?                 : Unknown
  Group (T)ype      :  S = Static, D = Dynamic
  Director (C)onfig :  F-S = Fibre-Switched, F-H = Fibre-Hub
```

```
                        G = GIGE, E = ESCON, T = T3, - = N/A
  Group Flags       :
      Prevent Auto (L)ink Recovery      : X = Enabled, . = Disabled
      Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
      Link (D)omino                     : X = Enabled, . = Disabled
      (S)TAR mode                       : N = Normal, R = Recovery, . = OFF
      RDF Software (C)ompression        : X = Enabled, . = Disabled, - = N/A
      RDF (H)ardware Compression        : X = Enabled, . = Disabled, - = N/A
      RDF Single Round (T)rip           : X = Enabled, . = Disabled, - = N/A
  RDFA Flags        :
      (C)onsistency : X = Enabled, . = Disabled, - = N/A
      (S)tatus      : A = Active, I = Inactive, - = N/A
      (R)DFA Mode   : S = Single-session, M = MSC, - = N/A
      (M)sc Cleanup : C = MSC Cleanup required, - = N/A
```

Example 4: Creating dynamic SRDF pairs

The commands in this example show how to create and verify dynamic SRDF pairs in a device group, set the SRDF mode for a device group, and then change the devices in these dynamic pairs to standard devices.

- The following `symdev list` command with the `-dynamic` option displays both R1 and R2 devices configured for dynamic SRDF capability. When combined with the `-r1` option, `symdev list -dynamic` displays devices configured for dynamic R1/R2 and R1-only; when combined with the `-r2` option, the command displays devices configured for dynamic R1/R2 and R2-only. The Config column shows these devices were not configured as SRDF devices.

```
‹R1Host#› symdev -sid 321 list -dynamic -r1 -r2

Symmetrix ID: 000192600321

        Device Name           Directors                   Device
--------------------------- ------------- -------------------------------------
                                                                          Cap
Sym  Physical               SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- -------------------------------------

0060 /dev/rhdisk27          10E:0 10A:D0  2-Way Mir     Grp'd        RW    4314
0061 /dev/rhdisk28          10E:0 09C:C0  2-Way Mir     Grp'd        RW    4314
0062 /dev/rhdisk29          10E:0 10B:C0  2-Way Mir     Grp'd        RW    4314
0063 /dev/rhdisk30          10E:0 09D:D0  2-Way Mir     Grp'd        RW    4314
0064 /dev/rhdisk31          10E:0 09A:C2  2-Way Mir     Grp'd        RW    4314
0065 /dev/rhdisk32          10E:0 10C:D2  2-Way Mir     Grp'd        RW    4314
0066 /dev/rhdisk33          10E:0 09B:D2  2-Way Mir     Grp'd        RW    4314
0067 /dev/rhdisk34          10E:0 10D:C2  2-Way Mir     Grp'd        RW    4314
```

- The following command uses the vi text editor to create the `devices` file consisting of dynamic SRDF pairs (R1/R2). Each R1 device in the first column has a corresponding R2 device in the second column. For example, the first line shows the pairing of 0060 (R1) with 0060 (R2).

```
‹R1Host#› vi devices
0060 0060
0061 0061
0062 0062
0063 0063
.
.
.
.
07EF 07EF
```

```
0800 0800
0801 0801
0802 0802
0803 0803
```

- The following `symrdf createpair` command establishes the dynamic pairs by parsing the `devices` file and specifying its column-1 devices as source devices on Symmetrix 321. Because this command is using SRDF group 128, communication is between Symmetrix 321 and remote Symmetrix 256. The `-invalidate r2` option invalidates all tracks on the R2 devices in preparation for a subsequent establish operation. The `-rdf_mode` option sets the mode to adaptive copy disk, which can transfer large amounts of data transfer without slowing performance. The `-g` option creates a device group named `drdf` and adds the dynamic SRDF pairs to the group.

```
‹R1Host#› symrdf -sid 321 -rdfg 128 -file devices createpair -invalidate r2 -nowd
   -type rdf1 -rdf_mode acp_disk -g drdf

An RDF 'Create Pair' operation execution is in progress for device
file 'devices'. Please wait...

    Create RDF Pair in (0321,128)....................................Started.
    Create RDF Pair in (0321,128)....................................Done.
    Mark target device(s) in (0321,128) for full copy from source....Started.
    Devices: 0060-0067, 07E0-07EF in (0321,128)..................... Marked.
    Devices: 0800-0803 in (0321,128)................................ Marked.
    Mark target device(s) in (0321,128) for full copy from source....Done.

The RDF 'Create Pair' operation successfully executed for device
file 'devices'.
```

- The following query shows the status of the dynamic SRDF pairs in the device group `drdf`. All pairs are in the Suspended state.

```
‹R1Host#› symrdf -g drdf query

Device Group (DG) Name             : drdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0    69030 NR 0060 WD       0        0 C.D.  Suspended
DEV002  0061 RW       0    69030 NR 0061 WD       0        0 C.D.  Suspended
DEV003  0062 RW       0    69030 NR 0062 WD       0        0 C.D.  Suspended
.
.
.
.
DEV026  0801 RW       0    69030 NR 0801 WD       0        0 C.D.  Suspended
DEV027  0802 RW       0    69030 NR 0802 WD       0        0 C.D.  Suspended
DEV028  0803 RW       0    69030 NR 0803 WD       0        0 C.D.  Suspended

Total          -------- --------           -------- --------
  Track(s)            0  1932840                  0        0
  MB(s)               0   120802                  0        0
```

- The `symrdf establish` command initiates copying R1 data to R2 devices. The `-invalidate r2` option from the previous command invalidated the R2 devices, a step that is usually carried out during a full establish operation. Consequently, you do not need the `-full` option. The invalidate step is not repeated, regardless of whether you use the `-full` option or not. If subsequently you re-establish or restore the dynamic SRDF pairs, omitting or including the `-full` option will affect how the copy occurs (either an incremental copy or a full copy, respectively). The output below says "Incremental Establish" because the `-full` option was omitted. However, because all tracks on the R2 devices were previously invalidated, the result is a full copy of all R1 tracks to the R2 tracks.

```
‹R1Host#› symrdf -g drdf establish -noprompt

An RDF 'Incremental Establish' operation execution is
in progress for device group 'drdf'. Please wait...

    Suspend RDF link(s).......................................Done.
    Resume RDF link(s).......................................Started.
    Resume RDF link(s).......................................Done.

The RDF 'Incremental Establish' operation successfully initiated for
device group 'drdf'.
```

- The following query displays the status of the dynamic SRDF pairs. The pairs are currently in the process of synchronizing (SyncInProg):

```
‹R1Host#› symrdf -g drdf query

Device Group (DG) Name             : drdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
              ST                  LI      ST
Standard       A                   N       A
Logical        T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev    E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0    68643 RW 0060 WD       0        0 C.D.  SyncInProg
DEV002  0061 RW       0    69030 RW 0061 WD       0        0 C.D.  SyncInProg
DEV003  0062 RW       0    67515 RW 0062 WD       0        0 C.D.  SyncInProg
.
.
.
DEV026  0801 RW       0    67965 RW 0801 WD       0        0 C.D.  SyncInProg
DEV027  0802 RW       0    68289 RW 0802 WD       0        0 C.D.  SyncInProg
DEV028  0803 RW       0    68205 RW 0803 WD       0        0 C.D.  SyncInProg

Total          -------- --------           -------- --------
  Track(s)            0  1896840                  0        0
  MB(s)               0   118552                  0        0
```

- The `symrdf verify` command with the `-synchronized` and `-summary` options provide complete details on whether the dynamic SRDF pairs in device group `drdf` have reached the Synchronized state.

```
‹R1Host#› symrdf -g drdf verify -synchronized -summary -i 15

Device Group (DG) Name : drdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                      0
  SyncInProg                       28
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            28

  RDF Mode                      Count
  --------------------------  ------
  Adaptive Copy Disk                28
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                       0
  -----------------------      ------
  Total                            28

                                 Track(s)    MB(s)
                               -----------  -------
  Total Source R1 Invalid                0        0
  Total Source R2 Invalid          1802979   112686

  Total Target R1 Invalid                0        0
  Total Target R2 Invalid                0        0

None of the devices in the group 'drdf' are in 'Synchronized' state.


Device Group (DG) Name : drdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     17
  SyncInProg                       11
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            28
```

```
  RDF Mode                      Count
  ---------------------------  ------
  Adaptive Copy Disk                28
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                       0
  -----------------------      ------
  Total                            28

                               Track(s)    MB(s)
                              -----------  -------
  Total Source R1 Invalid               0      0.0
  Total Source R2 Invalid           76884   4805.2

  Total Target R1 Invalid               0      0.0
  Total Target R2 Invalid               0      0.0

  Synchronization rate         :    158.8 MB/S
  Estimated time to completion : 00:00:21

Not All devices in the group 'drdf' are in 'Synchronized' state.

Device Group (DG) Name : drdf

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        0
  Synchronized                     28
  SyncInProg                        0
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                            28

  RDF Mode                      Count
  ---------------------------  ------
  Adaptive Copy Disk                28
  Adaptive Copy Write Pending       0
  Asynchronous                      0
  Semi-Synchronous                  0
   Synchronous                       0
  -----------------------      ------
  Total                            28

                               Track(s)    MB(s)
                              -----------  -------
  Total Source R1 Invalid               0      0.0
  Total Source R2 Invalid               0      0.0

  Total Target R1 Invalid               0      0.0
  Total Target R2 Invalid               0      0.0

All devices in the group 'drdf' are in 'Synchronized' state.
```

- The following query also confirms that all devices in `drdf` are in the Synchronized state:

‹R1Host#› **symrdf -g drdf query**

```
Device Group (DG) Name             : drdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 C.D.  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 C.D.  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 C.D.  Synchronized
DEV004  0063 RW       0        0 RW 0063 WD       0        0 C.D.  Synchronized
.
.
.
.
.
DEV026  0801 RW       0        0 RW 0801 WD       0        0 C.D.  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 C.D.  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 C.D.  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following `symrdf set mode` command sets the mode as synchronous for device group `drdf`:

  ‹R1Host#› **symrdf -g drdf set mode sync -noprompt**

  ```
  An RDF Set 'Synchronous Mode' operation execution is in
  progress for device group 'drdf'. Please wait...

  The RDF Set 'Synchronous Mode' operation successfully executed
  for device group 'drdf'.
  ```

- The following `symrdf split` command splits all dynamic SRDF pairs in device group `drdf`:

  ‹R1Host#› **symrdf -g drdf split -noprompt**

  ```
  An RDF 'Split' operation execution is
  in progress for device group 'drdf'. Please wait...

      Suspend RDF link(s).......................................Done.
      Read/Write Enable device(s) on RA at target (R2)..........Done.

  The RDF 'Split' operation successfully executed for
  device group 'drdf'.
  ```

- The following query confirms the SRDF pairs are in the Split state and the links are not ready (NR):

‹R1Host#› **symrdf -g drdf query**

```
Device Group (DG) Name             : drdf
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
```

```
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)

       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 RW       0        0 S...  Split
DEV002  0061 RW       0        0 NR 0061 RW       0        0 S...  Split
DEV003  0062 RW       0        0 NR 0062 RW       0        0 S...  Split
.
.
.
.
DEV026  0801 RW       0        0 NR 0801 RW       0        0 S...  Split
DEV027  0802 RW       0        0 NR 0802 RW       0        0 S...  Split
DEV028  0803 RW       0        0 NR 0803 RW       0        0 S...  Split

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- After the link state is not ready (NR), you can use the `symrdf deletepair` command to cancel the dynamic SRDF pairings defined in the device group, and delete this pairing information from the host's SYMAPI database file. This operation changes the type of the device group from RDF1 to REGULAR and the R1 devices in the device group change to standard devices.

```
‹R1Host#› symrdf -g drdf deletepair -noprompt

An RDF 'Delete Pair' operation execution is
in progress for device group 'drdf'. Please wait...

    Delete RDF Pair.........................................Started.
    Delete RDF Pair.........................................Done.

The RDF 'Delete Pair' operation successfully executed for
device group 'drdf'.
```

- The following `symrdf query` confirms that `drdf` does not have any associated SRDF devices:

```
‹R1Host#› symrdf -g drdf query

Device Group 'drdf' has no associated RDF devices that match the
criteria specified.
```

- The `symdg list ld` command shows the DG's Type for `drdf` is now REGULAR:

```
‹R1Host#› symdg -g drdf list ld

Device Group (DG) Name: drdf
DG's Type              : REGULAR
DG's Symmetrix ID      : 000192600321


        Standard Device Name           Directors              Device
----------------------------------   ------------- ----------------------------
                                                                          Cap
Logical     Physical          Sym    SA :P DA :IT  Config        Att Sts   (MB)
----------------------------------   ------------- ----------------------------
```

```
DEV001      /dev/rhdisk27     0060   10E:0 10A:D0  2-Way Mir         RW    4314
DEV002      /dev/rhdisk28     0061   10E:0 09C:C0  2-Way Mir         RW    4314
DEV003      /dev/rhdisk29     0062   10E:0 10B:C0  2-Way Mir         RW    4314
DEV004      /dev/rhdisk30     0063   10E:0 09D:D0  2-Way Mir         RW    4314

.
.
.
.
DEV026      /dev/rhdisk216    0801   10E:0  NA:NA  TDEV              RW    4314
DEV027      /dev/rhdisk217    0802   10E:0  NA:NA  TDEV              RW    4314
DEV028      /dev/rhdisk218    0803   10E:0  NA:NA  TDEV              RW    4314
```

Example 5: Operating with SRDF asynchronous replication

The commands in this example use a source Symmetrix array (sid 321) connected to a target Symmetrix array (sid 256) through the SRDF (RA) group number 128.

- The `symrdf list` command with the `-rdfa` option displays all devices configured for SRDF/A operations on Symmetrix 321. The RDF Type:G column shows that SRDF group number 128 is an SRDF/A group. Devices in this group type can either be all R1 devices or all R2 devices but not both. In this listing, they are all R1 devices. Note that starting with Enginuity 5671, all SRDF groups have the ability to operate in SRDF/A mode.

```
‹R1Host#› symrdf -sid 321 list -rdfa

Symmetrix ID: 000192600321

                              Local Device View
---------------------------------------------------------------------------
                     STATUS     MODES                      RDF  S T A T E S
Sym        RDF      ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G    SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------  ---------  ----- -------  ------- --- ---- -------------

0060 0060   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0061 0061   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0062 0062   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0063 0063   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0064 0064   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0065 0065   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0066 0066   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0067 0067   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E0 07E0   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E1 07E1   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E2 07E2   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E3 07E3   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E4 07E4   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E5 07E5   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E6 07E6   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E7 07E7   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E8 07E8   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07E9 07E9   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EA 07EA   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EB 07EB   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EC 07EC   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07ED 07ED   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EE 07EE   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
07EF 07EF   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0800 0800   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0801 0801   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
```

```
0802 0802   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized
0803 0803   R1:128  RW RW RW   C.D1.       0        0 RW  WD   Synchronized

Total                                 -------- --------
  Track(s)                                   0        0
  MB(s)                                    0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Mirror) T(ype)       : 1 = R1, 2 = R2
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symdg create` command creates an RDF1 type device group named `AsyncGrp1`. The following `symdg addall` command adds all devices from SRDF group `128` to `AsyncGrp1`. For asynchronous replication, all devices in an SRDF group must be managed together.

```
‹R1Host#› symdg create AsyncGrp1 -type rdf1
‹R1Host#› symdg -g AsyncGrp1 -sid 321 addall dev -sel_rdfg 128 -v

Device(s) 00060:00067: ADD DEVS BY RANGE Succeeded.

Device(s) 007E0:007EF: ADD DEVS BY RANGE Succeeded.

Device(s) 00800:00803: ADD DEVS BY RANGE Succeeded.
```

- The `query` displays the status of the SRDF pairs in `AsyncGrp1`. Currently the pairs are in **Synchronized** state and running with adaptive copy (**C**) replication.

```
‹R1Host#› symrdf -g AsyncGrp1 query

Device Group (DG) Name             : AsyncGrp1
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 C.D.  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 C.D.  Synchronized
DEV003  0062 RW       0        0 RW 0062 WD       0        0 C.D.  Synchronized
DEV004  0063 RW       0        0 RW 0063 WD       0        0 C.D.  Synchronized
DEV005  0064 RW       0        0 RW 0064 WD       0        0 C.D.  Synchronized
DEV006  0065 RW       0        0 RW 0065 WD       0        0 C.D.  Synchronized
DEV007  0066 RW       0        0 RW 0066 WD       0        0 C.D.  Synchronized
DEV008  0067 RW       0        0 RW 0067 WD       0        0 C.D.  Synchronized
DEV009  07E0 RW       0        0 RW 07E0 WD       0        0 C.D.  Synchronized
DEV010  07E1 RW       0        0 RW 07E1 WD       0        0 C.D.  Synchronized
DEV011  07E2 RW       0        0 RW 07E2 WD       0        0 C.D.  Synchronized
DEV012  07E3 RW       0        0 RW 07E3 WD       0        0 C.D.  Synchronized
DEV013  07E4 RW       0        0 RW 07E4 WD       0        0 C.D.  Synchronized
DEV014  07E5 RW       0        0 RW 07E5 WD       0        0 C.D.  Synchronized
DEV015  07E6 RW       0        0 RW 07E6 WD       0        0 C.D.  Synchronized
```

```
DEV016  07E7 RW       0        0 RW 07E7 WD       0        0 C.D.  Synchronized
DEV017  07E8 RW       0        0 RW 07E8 WD       0        0 C.D.  Synchronized
DEV018  07E9 RW       0        0 RW 07E9 WD       0        0 C.D.  Synchronized
DEV019  07EA RW       0        0 RW 07EA WD       0        0 C.D.  Synchronized
DEV020  07EB RW       0        0 RW 07EB WD       0        0 C.D.  Synchronized
DEV021  07EC RW       0        0 RW 07EC WD       0        0 C.D.  Synchronized
DEV022  07ED RW       0        0 RW 07ED WD       0        0 C.D.  Synchronized
DEV023  07EE RW       0        0 RW 07EE WD       0        0 C.D.  Synchronized
DEV024  07EF RW       0        0 RW 07EF WD       0        0 C.D.  Synchronized
DEV025  0800 RW       0        0 RW 0800 WD       0        0 C.D.  Synchronized
DEV026  0801 RW       0        0 RW 0801 WD       0        0 C.D.  Synchronized
DEV027  0802 RW       0        0 RW 0802 WD       0        0 C.D.  Synchronized
DEV028  0803 RW       0        0 RW 0803 WD       0        0 C.D.  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Mirror) T(ype)       : 1 = R1, 2 = R2
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf set mode async` command sets the method of replication to asynchronous for all devices in `AsyncGrp1`. At this point, SRDF consistency protection is not enabled.

  ```
  ‹R1Host#› symrdf -g AsyncGrp1 set mode async -noprompt

  An RDF Set 'Asynchronous Mode' operation execution is in
  progress for device group 'AsyncGrp1'. Please wait...

  The RDF Set 'Asynchronous Mode' operation successfully executed
  for device group 'AsyncGrp1'.
  ```

- The following `symrdf enable` command enables consistency protection for devices in `AsyncGrp1`:

  ```
  ‹R1Host#› symrdf -g AsyncGrp1 enable -noprompt

  An RDF 'Enable' operation execution is
  in progress for device group 'AsyncGrp1'. Please wait...

  The RDF 'Enable' operation successfully executed for
  device group 'AsyncGrp1'.
  ```

- The following `symdg show` command verifies the SRDF/A session is active and the consistency state is enabled, as shown in the RDFA Information section:

```
‹R1Host#› symdg show AsyncGrp1

Group Name:  AsyncGrp1

    Group Type                                  : RDF1     (RDFA)
    Device Group in GNS                         : Yes
    Valid                                       : Yes
    Symmetrix ID                                : 000192600321
    Group Creation Time                         : Fri Jul 23 15:02:22 2010
    Vendor ID                                   : EMC Corp
    Application ID                              : SYMCLI

    Number of STD Devices in Group              :   28
    Number of Associated GK's                   :    0
```

```
Number of Locally-associated BCV's           :    0
Number of Locally-associated VDEV's          :    0
Number of Locally-associated TGT's           :    0
Number of Remotely-associated VDEV's(STD RDF):    0
Number of Remotely-associated BCV's (STD RDF):    0
Number of Remotely-associated TGT's(TGT RDF) :    0
Number of Remotely-associated BCV's (BCV RDF):    0
Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0
Number of Composite Groups                   :    0
Composite Group Names                        : N/A

Standard (STD) Devices (28):
    {
    ------------------------------------------------------------------------------
                                                Sym  Device                     Cap
    LdevName              PdevName              Dev  Config        Att. Sts     (MB)
    ------------------------------------------------------------------------------
    DEV001                /dev/rhdisk27         0060 RDF1+Mir           RW      4314
    DEV002                /dev/rhdisk28         0061 RDF1+Mir           RW      4314
    DEV003                /dev/rhdisk29         0062 RDF1+Mir           RW      4314
    DEV004                /dev/rhdisk30         0063 RDF1+Mir           RW      4314
    DEV005                /dev/rhdisk31         0064 RDF1+Mir           RW      4314
    DEV006                /dev/rhdisk32         0065 RDF1+Mir           RW      4314
    DEV007                /dev/rhdisk33         0066 RDF1+Mir           RW      4314
    DEV008                /dev/rhdisk34         0067 RDF1+Mir           RW      4314
    DEV009                /dev/rhdisk43         07E0 RDF1+TDEV          RW      4314
    DEV010                /dev/rhdisk44         07E1 RDF1+TDEV          RW      4314
    DEV011                /dev/rhdisk45         07E2 RDF1+TDEV          RW      4314
    DEV012                /dev/rhdisk46         07E3 RDF1+TDEV          RW      4314
    DEV013                /dev/rhdisk47         07E4 RDF1+TDEV          RW      4314
    DEV014                /dev/rhdisk48         07E5 RDF1+TDEV          RW      4314
    DEV015                /dev/rhdisk49         07E6 RDF1+TDEV          RW      4314
    DEV016                /dev/rhdisk50         07E7 RDF1+TDEV          RW      4314
    DEV017                /dev/rhdisk51         07E8 RDF1+TDEV          RW      4314
    DEV018                /dev/rhdisk52         07E9 RDF1+TDEV          RW      4314
    DEV019                /dev/rhdisk53         07EA RDF1+TDEV          RW      4314
    DEV020                /dev/rhdisk54         07EB RDF1+TDEV          RW      4314
    DEV021                /dev/rhdisk55         07EC RDF1+TDEV          RW      4314
    DEV022                /dev/rhdisk56         07ED RDF1+TDEV          RW      4314
    DEV023                /dev/rhdisk57         07EE RDF1+TDEV          RW      4314
    DEV024                /dev/rhdisk58         07EF RDF1+TDEV          RW      4314
    DEV025                /dev/rhdisk76         0800 RDF1+TDEV          RW      4314
    DEV026                /dev/rhdisk216        0801 RDF1+TDEV          RW      4314
    DEV027                /dev/rhdisk217        0802 RDF1+TDEV          RW      4314
    DEV028                /dev/rhdisk218        0803 RDF1+TDEV          RW      4314
    }
```

Device Group RDF Information

```
    {
    RDF Type                                : R1
    RDF (RA) Group Number                   : 128               (7F)

    Remote Symmetrix ID                     : 000192600256

    R2 Device Is Larger Than The R1 Device : False

    Paired with a Diskless Device           : False
    Paired with a Concurrent Device         : False
    Paired with a Cascaded Device           : False
    Thick Thin Relationship                 : False

    RDF Pair Configuration                  : Normal
    RDF STAR Mode                           : False
```

```
RDF Mode                                : Asynchronous
RDF Adaptive Copy                       : Disabled
RDF Adaptive Copy Write Pending State   : N/A
RDF Adaptive Copy Skew (Tracks)         : 32767

RDF Device Domino                       : Disabled

RDF Link Configuration                  : Fibre
RDF Link Domino                         : Disabled
Prevent Automatic RDF Link Recovery     : Enabled
Prevent RAs Online Upon Power ON        : Enabled

Device RDF Status                       : Ready           (RW)

Device RA Status                        : Ready           (RW)
Device Link Status                      : Ready           (RW)
Time of Last Device Link Status Change  : N/A

Device Suspend State                    : N/A
Device Consistency State                : Enabled
Device Consistency Exempt State         : Disabled
RDF R2 Not Ready If Invalid             : Disabled
Device Write Pacing Exempt State        : Disabled
Effective Write Pacing Exempt State     : Disabled

Device RDF State                        : Ready           (RW)
Remote Device RDF State                 : Write Disabled  (WD)

RDF Pair State (  R1 <===> R2 )         : Consistent

Number of R1 Invalid Tracks             : 0
Number of R2 Invalid Tracks             : 0

RDFA Information:
    {
    Session Number                      : 127
    Cycle Number                        : 3
    Number of Devices in the Session    : 28
    Session Status                      : Active
    Consistency Exempt Devices          : No
    Write Pacing Exempt Devices         : No

    Session Consistency State           : Enabled
    Minimum Cycle Time                  : 00:00:30
    Average Cycle Time                  : 00:01:00
    Duration of Last cycle              : 00:00:30
    Session Priority                    : 33

    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1           : 00:00:43
    R2 Image Capture Time               : Fri Jul 23 15:04:23 2010
    R2 Data is Consistent               : True
    R1 Side Percent Cache In Use        :  0
    R2 Side Percent Cache In Use        :  0

    Transmit Idle Time                  : 00:00:00
    R1 Side DSE Used Tracks             :  0
    R2 Side DSE Used Tracks             :  0
    R1 Side Shared Tracks               : 0
    }
}
```

- The following `symrdf verify` with the `-consistent` option shows the SRDF pairs in `AsyncGrp1ar` are in the Consistent state:

```
‹R1Host#› symrdf -g AsyncGrp1 verify -consistent

All devices in the group 'AsyncGrp1' are in 'Consistent' state.
```

- The following `symrdf suspend` command with the `-force` option trips the device group, making the devices not ready (NR) on the link. This operation is useful if you need to trip the device group but also maintain the consistency of the R2 database copy with the production copy on the R1 side. The `-force` option is required to ensure that you want to stop SRDF/A operations and end consistency protection.

```
‹R1Host#› symrdf -g AsyncGrp1 suspend -force -noprompt

An RDF 'Suspend' operation execution is
in progress for device group 'AsyncGrp1'. Please wait...

    Suspend RDF link(s).....................................Started.
    Suspend RDF link(s).....................................Done.

The RDF 'Suspend' operation successfully executed for
device group 'AsyncGrp1'.
```

- The following `symrdf query` command with the `-rdfa` option shows the SRDF/A session status is now **Inactive** and that the SRDF pairs are in the Suspended state. Typically, you would see invalid tracks on the R1 side to indicate continuing I/O on the R1 side, but there is no I/O in this example.

```
‹R1Host#› symrdf -g AsyncGrp1 query -rdfa

Device Group (DG) Name             : AsyncGrp1
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)

RDFA Session Number                : 127
RDFA Cycle Number                  : 0
RDFA Session Status                : Inactive
RDFA Consistency Exempt Devices    : No
RDFA Minimum Cycle Time            : 00:00:30
RDFA Avg Cycle Time                : 00:00:00
Duration of Last cycle             : 00:00:00
RDFA Session Priority              : 33
Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1          : 00:00:54
R2 Image Capture Time              : Fri Jul 23 15:05:53 2010
R2 Data is Consistent              : True
RDFA R1 Side Percent Cache In Use  :  0
RDFA R2 Side Percent Cache In Use  :  0
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
Transmit Idle Time                 : 00:00:00
R1 Side Shared Tracks              : 0

Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDACE STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 NR 0060 WD       0        0 A..X. Suspended
```

```
DEV002  0061 RW       0         0 NR 0061 WD       0         0 A..X. Suspended
DEV003  0062 RW       0         0 NR 0062 WD       0         0 A..X. Suspended
DEV004  0063 RW       0         0 NR 0063 WD       0         0 A..X. Suspended
DEV005  0064 RW       0         0 NR 0064 WD       0         0 A..X. Suspended
DEV006  0065 RW       0         0 NR 0065 WD       0         0 A..X. Suspended
DEV007  0066 RW       0         0 NR 0066 WD       0         0 A..X. Suspended
DEV008  0067 RW       0         0 NR 0067 WD       0         0 A..X. Suspended
DEV009  07E0 RW       0         0 NR 07E0 WD       0         0 A..X. Suspended
DEV010  07E1 RW       0         0 NR 07E1 WD       0         0 A..X. Suspended
DEV011  07E2 RW       0         0 NR 07E2 WD       0         0 A..X. Suspended
DEV012  07E3 RW       0         0 NR 07E3 WD       0         0 A..X. Suspended
DEV013  07E4 RW       0         0 NR 07E4 WD       0         0 A..X. Suspended
DEV014  07E5 RW       0         0 NR 07E5 WD       0         0 A..X. Suspended
DEV015  07E6 RW       0         0 NR 07E6 WD       0         0 A..X. Suspended
DEV016  07E7 RW       0         0 NR 07E7 WD       0         0 A..X. Suspended
DEV017  07E8 RW       0         0 NR 07E8 WD       0         0 A..X. Suspended
DEV018  07E9 RW       0         0 NR 07E9 WD       0         0 A..X. Suspended
DEV019  07EA RW       0         0 NR 07EA WD       0         0 A..X. Suspended
DEV020  07EB RW       0         0 NR 07EB WD       0         0 A..X. Suspended
DEV021  07EC RW       0         0 NR 07EC WD       0         0 A..X. Suspended
DEV022  07ED RW       0         0 NR 07ED WD       0         0 A..X. Suspended
DEV023  07EE RW       0         0 NR 07EE WD       0         0 A..X. Suspended
DEV024  07EF RW       0         0 NR 07EF WD       0         0 A..X. Suspended
DEV025  0800 RW       0         0 NR 0800 WD       0         0 A..X. Suspended
DEV026  0801 RW       0         0 NR 0801 WD       0         0 A..X. Suspended
DEV027  0802 RW       0         0 NR 0802 WD       0         0 A..X. Suspended
DEV028  0803 RW       0         0 NR 0803 WD       0         0 A..X. Suspended

Total          -------- --------           -------- --------
  Track(s)            0        0                   0        0
  MB(s)             0.0      0.0                 0.0      0.0
```

- The following `symrdf resume` command resumes the SRDF links between the SRDF pairs in `AsyncGrp1` and I/O traffic between the R1 devices and their paired R2 devices. The SRDF/A session is automatically activated again.

```
‹R1Host#› symrdf -g AsyncGrp1 resume -noprompt

An RDF 'Resume' operation execution is
in progress for device group 'AsyncGrp1'. Please wait...

    Resume RDF link(s)........................................Started.
    Resume RDF link(s)........................................Done.

The RDF 'Resume' operation successfully executed for
device group 'AsyncGrp1'.
```

- At this point, the SRDF/A devices are ready again on the SRDF link and operating with asynchronous replication, and consistency protection remains set. The following query verifies the SRDF/A session is active again and the devices are in the Consistent state:

```
‹R1Host#› symrdf -g AsyncGrp1 query -rdfa

Device Group (DG) Name             : AsyncGrp1
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)

RDFA Session Number                : 127
RDFA Cycle Number                  : 2
RDFA Session Status                : Active
RDFA Consistency Exempt Devices    : No
RDFA Minimum Cycle Time            : 00:00:30
RDFA Avg Cycle Time                : 00:00:30
Duration of Last cycle             : 00:00:30
```

```
RDFA Session Priority              : 33
Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1          : 00:00:47
R2 Image Capture Time              : Fri Jul 23 15:07:34 2010
R2 Data is Consistent              : True
RDFA R1 Side Percent Cache In Use  :  0
RDFA R2 Side Percent Cache In Use  :  0
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
Transmit Idle Time                 : 00:00:00
R1 Side Shared Tracks              : 0

Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDACE STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 A..X. Consistent
DEV002  0061 RW       0        0 RW 0061 WD       0        0 A..X. Consistent
DEV003  0062 RW       0        0 RW 0062 WD       0        0 A..X. Consistent
DEV004  0063 RW       0        0 RW 0063 WD       0        0 A..X. Consistent
DEV005  0064 RW       0        0 RW 0064 WD       0        0 A..X. Consistent
DEV006  0065 RW       0        0 RW 0065 WD       0        0 A..X. Consistent
DEV007  0066 RW       0        0 RW 0066 WD       0        0 A..X. Consistent
DEV008  0067 RW       0        0 RW 0067 WD       0        0 A..X. Consistent
DEV009  07E0 RW       0        0 RW 07E0 WD       0        0 A..X. Consistent
DEV010  07E1 RW       0        0 RW 07E1 WD       0        0 A..X. Consistent
DEV011  07E2 RW       0        0 RW 07E2 WD       0        0 A..X. Consistent
DEV012  07E3 RW       0        0 RW 07E3 WD       0        0 A..X. Consistent
DEV013  07E4 RW       0        0 RW 07E4 WD       0        0 A..X. Consistent
DEV014  07E5 RW       0        0 RW 07E5 WD       0        0 A..X. Consistent
DEV015  07E6 RW       0        0 RW 07E6 WD       0        0 A..X. Consistent
DEV016  07E7 RW       0        0 RW 07E7 WD       0        0 A..X. Consistent
DEV017  07E8 RW       0        0 RW 07E8 WD       0        0 A..X. Consistent
DEV018  07E9 RW       0        0 RW 07E9 WD       0        0 A..X. Consistent
DEV019  07EA RW       0        0 RW 07EA WD       0        0 A..X. Consistent
DEV020  07EB RW       0        0 RW 07EB WD       0        0 A..X. Consistent
DEV021  07EC RW       0        0 RW 07EC WD       0        0 A..X. Consistent
DEV022  07ED RW       0        0 RW 07ED WD       0        0 A..X. Consistent
DEV023  07EE RW       0        0 RW 07EE WD       0        0 A..X. Consistent
DEV024  07EF RW       0        0 RW 07EF WD       0        0 A..X. Consistent
DEV025  0800 RW       0        0 RW 0800 WD       0        0 A..X. Consistent
DEV026  0801 RW       0        0 RW 0801 WD       0        0 A..X. Consistent
DEV027  0802 RW       0        0 RW 0802 WD       0        0 A..X. Consistent
DEV028  0803 RW       0        0 RW 0803 WD       0        0 A..X. Consistent

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

Example 6: Using a composite group to control SRDF pairs

The commands in this example illustrate how to create a composite group, add a range of devices to it, perform an incremental establish on it, enable consistency protection to it, check the state of its SRDF pairs, and suspend the operations on it. For more examples on using SRDF consistency protection, refer to Chapter 15, "Implementing Consistency Protection,".

- The following `symcg create` command creates the `SRDF` composite group of RDF1 type with consistency enabled:

```
‹R1Host#› symcg create SRDF -type rdf1 -rdf_consistency
```

- The following `symcg addall` commands add a range of standard devices on Symmetrix 321 to the `SRDF` composite group:

```
‹R1Host#› symcg -cg SRDF -sid 321 addall dev -devs 60:67,7e0:7ef,800:803 -v

Device(s) 00060:00067: ADD DEVS BY RANGE Succeeded.

Device(s) 007E0:007EF: ADD DEVS BY RANGE Succeeded.

Device(s) 00800:00803: ADD DEVS BY RANGE Succeeded.

‹R1Host#› symcg -cg SRDF -sid 321 addall dev -devs 808:817,828:833 -v

Device(s) 00808:00817: ADD DEVS BY RANGE Succeeded.

Device(s) 00828:00833: ADD DEVS BY RANGE Succeeded.
```

- The following query checks the state of the SRDF pairs in the `SRDF` composite group. Note that the SRDF pairs on one Symmetrix are in Suspended state, while the pairs in the other Symmetrix array are in Synchronized state.

```
‹R1Host#› symrdf -cg SRDF query

Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number             : 128 (7F)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
------------------------------- -- ------------------------- ----- ------ ------------
DEV001   0060 RW       0       0 NR 0060 WD       0       0 S...  .  .   Suspended
DEV002   0061 RW       0       0 NR 0061 WD       0       0 S...  .  .   Suspended
DEV003   0062 RW       0       0 NR 0062 WD       0       0 S...  .  .   Suspended
DEV004   0063 RW       0       0 NR 0063 WD       0       0 S...  .  .   Suspended
DEV005   0064 RW       0       0 NR 0064 WD       0       0 S...  .  .   Suspended
DEV006   0065 RW       0       0 NR 0065 WD       0       0 S...  .  .   Suspended
DEV007   0066 RW       0       0 NR 0066 WD       0       0 S...  .  .   Suspended
DEV008   0067 RW       0       0 NR 0067 WD       0       0 S...  .  .   Suspended
DEV009   07E0 RW       0       0 NR 07E0 WD       0       0 S...  .  .   Suspended
```

```
DEV010   07E1 RW       0       0 NR 07E1 WD       0       0 S...  .  .   Suspended
DEV011   07E2 RW       0       0 NR 07E2 WD       0       0 S...  .  .   Suspended
DEV012   07E3 RW       0       0 NR 07E3 WD       0       0 S...  .  .   Suspended
DEV013   07E4 RW       0       0 NR 07E4 WD       0       0 S...  .  .   Suspended
DEV014   07E5 RW       0       0 NR 07E5 WD       0       0 S...  .  .   Suspended
DEV015   07E6 RW       0       0 NR 07E6 WD       0       0 S...  .  .   Suspended
DEV016   07E7 RW       0       0 NR 07E7 WD       0       0 S...  .  .   Suspended
DEV017   07E8 RW       0       0 NR 07E8 WD       0       0 S...  .  .   Suspended
DEV018   07E9 RW       0       0 NR 07E9 WD       0       0 S...  .  .   Suspended
DEV019   07EA RW       0       0 NR 07EA WD       0       0 S...  .  .   Suspended
DEV020   07EB RW       0       0 NR 07EB WD       0       0 S...  .  .   Suspended
DEV021   07EC RW       0       0 NR 07EC WD       0       0 S...  .  .   Suspended
DEV022   07ED RW       0       0 NR 07ED WD       0       0 S...  .  .   Suspended
DEV023   07EE RW       0       0 NR 07EE WD       0       0 S...  .  .   Suspended
DEV024   07EF RW       0       0 NR 07EF WD       0       0 S...  .  .   Suspended
DEV025   0800 RW       0       0 NR 0800 WD       0       0 S...  .  .   Suspended
DEV026   0801 RW       0       0 NR 0801 WD       0       0 S...  .  .   Suspended
DEV027   0802 RW       0       0 NR 0802 WD       0       0 S...  .  .   Suspended
DEV028   0803 RW       0       0 NR 0803 WD       0       0 S...  .  .   Suspended

Symmetrix ID                     : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID              : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number            : 129 (80)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV029   0808 RW       0       0 RW 0808 WD       0       0 C.D.  .  -   Synchronized
DEV030   0809 RW       0       0 RW 0809 WD       0       0 C.D.  .  -   Synchronized
DEV031   080A RW       0       0 RW 080A WD       0       0 C.D.  .  -   Synchronized
DEV032   080B RW       0       0 RW 080B WD       0       0 C.D.  .  -   Synchronized
DEV033   080C RW       0       0 RW 080C WD       0       0 C.D.  .  -   Synchronized
DEV034   080D RW       0       0 RW 080D WD       0       0 C.D.  .  -   Synchronized
DEV035   080E RW       0       0 RW 080E WD       0       0 C.D.  .  -   Synchronized
DEV036   080F RW       0       0 RW 080F WD       0       0 C.D.  .  -   Synchronized
DEV037   0810 RW       0       0 RW 0810 WD       0       0 C.D.  .  -   Synchronized
DEV038   0811 RW       0       0 RW 0811 WD       0       0 C.D.  .  -   Synchronized
DEV039   0812 RW       0       0 RW 0812 WD       0       0 C.D.  .  -   Synchronized
DEV040   0813 RW       0       0 RW 0813 WD       0       0 C.D.  .  -   Synchronized
DEV041   0814 RW       0       0 RW 0814 WD       0       0 C.D.  .  -   Synchronized
DEV042   0815 RW       0       0 RW 0815 WD       0       0 C.D.  .  -   Synchronized
DEV043   0816 RW       0       0 RW 0816 WD       0       0 C.D.  .  -   Synchronized
DEV044   0817 RW       0       0 RW 0817 WD       0       0 C.D.  .  -   Synchronized
DEV045   0828 RW       0       0 RW 0828 WD       0       0 C.D.  .  -   Synchronized
DEV046   0829 RW       0       0 RW 0829 WD       0       0 C.D.  .  -   Synchronized
DEV047   082A RW       0       0 RW 082A WD       0       0 C.D.  .  -   Synchronized
DEV048   082B RW       0       0 RW 082B WD       0       0 C.D.  .  -   Synchronized
DEV049   082C RW       0       0 RW 082C WD       0       0 C.D.  .  -   Synchronized
DEV050   082D RW       0       0 RW 082D WD       0       0 C.D.  .  -   Synchronized
DEV051   082E RW       0       0 RW 082E WD       0       0 C.D.  .  -   Synchronized
DEV052   082F RW       0       0 RW 082F WD       0       0 C.D.  .  -   Synchronized
DEV053   0830 RW       0       0 RW 0830 WD       0       0 C.D.  .  -   Synchronized
DEV054   0831 RW       0       0 RW 0831 WD       0       0 C.D.  .  -   Synchronized
DEV055   0832 RW       0       0 RW 0832 WD       0       0 C.D.  .  -   Synchronized
DEV056   0833 RW       0       0 RW 0833 WD       0       0 C.D.  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:
```

```
 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following `symrdf establish` command initiates an incremental establish operation on the pairs in the composite group that are not synchronized (that is, the suspended pairs in Symmetrix 321):

```
‹R1Host#› symrdf -cg SRDF establish -noprompt

An RDF 'Incremental Establish' operation execution is
in progress for composite group 'SRDF'. Please wait...

    Suspend RDF link(s) for device(s) in (0321,128)..................Done.
    Resume RDF link(s) for device(s) in (0321,128)...................Started.
    Resume RDF link(s) for device(s) in (0321,128)..................Done.

The RDF 'Incremental Establish' operation successfully initiated for
composite group 'SRDF'.
```

- The following query checks the state of the pairs of the `SRDF` composite group, and shows that the previously suspended pairs are completing synchronization:

```
‹R1Host#› symrdf -cg SRDF query

Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number             : 128 (7F)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ------------------------ ----- ------ ------------
DEV001   0060 RW       0    8192 RW 0060 WD       0       0 S...  .  -   SyncInProg
DEV002   0061 RW       0       0 RW 0061 WD       0       0 S...  .  -   Synchronized
DEV003   0062 RW       0       0 RW 0062 WD       0       0 S...  .  -   Synchronized
DEV004   0063 RW       0       0 RW 0063 WD       0       0 S...  .  -   Synchronized
DEV005   0064 RW       0       0 RW 0064 WD       0       0 S...  .  -   Synchronized
DEV006   0065 RW       0       0 RW 0065 WD       0       0 S...  .  -   Synchronized
DEV007   0066 RW       0       0 RW 0066 WD       0       0 S...  .  -   Synchronized
DEV008   0067 RW       0       0 RW 0067 WD       0       0 S...  .  -   Synchronized
DEV009   07E0 RW       0       0 RW 07E0 WD       0       0 S...  .  -   Synchronized
DEV010   07E1 RW       0       0 RW 07E1 WD       0       0 S...  .  -   Synchronized
DEV011   07E2 RW       0       0 RW 07E2 WD       0       0 S...  .  -   Synchronized
DEV012   07E3 RW       0       0 RW 07E3 WD       0       0 S...  .  -   Synchronized
DEV013   07E4 RW       0       0 RW 07E4 WD       0       0 S...  .  -   Synchronized
DEV014   07E5 RW       0       0 RW 07E5 WD       0       0 S...  .  -   Synchronized
DEV015   07E6 RW       0       0 RW 07E6 WD       0       0 S...  .  -   Synchronized
DEV016   07E7 RW       0       0 RW 07E7 WD       0       0 S...  .  -   Synchronized
```

```
DEV017   07E8 RW       0       0 RW 07E8 WD       0       0 S...  .  -   Synchronized
DEV018   07E9 RW       0       0 RW 07E9 WD       0       0 S...  .  -   Synchronized
DEV019   07EA RW       0       0 RW 07EA WD       0       0 S...  .  -   Synchronized
DEV020   07EB RW       0       0 RW 07EB WD       0       0 S...  .  -   Synchronized
DEV021   07EC RW       0       0 RW 07EC WD       0       0 S...  .  -   Synchronized
DEV022   07ED RW       0       0 RW 07ED WD       0       0 S...  .  -   Synchronized
DEV023   07EE RW       0       0 RW 07EE WD       0       0 S...  .  -   Synchronized
DEV024   07EF RW       0       0 RW 07EF WD       0       0 S...  .  -   Synchronized
DEV025   0800 RW       0       0 RW 0800 WD       0       0 S...  .  -   Synchronized
DEV026   0801 RW       0       0 RW 0801 WD       0       0 S...  .  -   Synchronized
DEV027   0802 RW       0       0 RW 0802 WD       0       0 S...  .  -   Synchronized
DEV028   0803 RW       0       0 RW 0803 WD       0       0 S...  .  -   Synchronized

Symmetrix ID                      : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number             : 129 (80)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------ ----- ------ ------------
               ST                 LI      ST                      C  S
  Standard     A                   N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv   K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks    S Dev   E  Tracks  Tracks MDAE  s  p   STATE
--------------------------------  -- ------------------------ ----- ------ ------------
DEV029   0808 RW       0       0 RW 0808 WD       0       0 C.D.  .  -   Synchronized
DEV030   0809 RW       0       0 RW 0809 WD       0       0 C.D.  .  -   Synchronized
DEV031   080A RW       0       0 RW 080A WD       0       0 C.D.  .  -   Synchronized
DEV032   080B RW       0       0 RW 080B WD       0       0 C.D.  .  -   Synchronized
DEV033   080C RW       0       0 RW 080C WD       0       0 C.D.  .  -   Synchronized
DEV034   080D RW       0       0 RW 080D WD       0       0 C.D.  .  -   Synchronized
DEV035   080E RW       0       0 RW 080E WD       0       0 C.D.  .  -   Synchronized
DEV036   080F RW       0       0 RW 080F WD       0       0 C.D.  .  -   Synchronized
DEV037   0810 RW       0       0 RW 0810 WD       0       0 C.D.  .  -   Synchronized
DEV038   0811 RW       0       0 RW 0811 WD       0       0 C.D.  .  -   Synchronized
DEV039   0812 RW       0       0 RW 0812 WD       0       0 C.D.  .  -   Synchronized
DEV040   0813 RW       0       0 RW 0813 WD       0       0 C.D.  .  -   Synchronized
DEV041   0814 RW       0       0 RW 0814 WD       0       0 C.D.  .  -   Synchronized
DEV042   0815 RW       0       0 RW 0815 WD       0       0 C.D.  .  -   Synchronized
DEV043   0816 RW       0       0 RW 0816 WD       0       0 C.D.  .  -   Synchronized
DEV044   0817 RW       0       0 RW 0817 WD       0       0 C.D.  .  -   Synchronized
DEV045   0828 RW       0       0 RW 0828 WD       0       0 C.D.  .  -   Synchronized
DEV046   0829 RW       0       0 RW 0829 WD       0       0 C.D.  .  -   Synchronized
DEV047   082A RW       0       0 RW 082A WD       0       0 C.D.  .  -   Synchronized
DEV048   082B RW       0       0 RW 082B WD       0       0 C.D.  .  -   Synchronized
DEV049   082C RW       0       0 RW 082C WD       0       0 C.D.  .  -   Synchronized
DEV050   082D RW       0       0 RW 082D WD       0       0 C.D.  .  -   Synchronized
DEV051   082E RW       0       0 RW 082E WD       0       0 C.D.  .  -   Synchronized
DEV052   082F RW       0       0 RW 082F WD       0       0 C.D.  .  -   Synchronized
DEV053   0830 RW       0       0 RW 0830 WD       0       0 C.D.  .  -   Synchronized
DEV054   0831 RW       0       0 RW 0831 WD       0       0 C.D.  .  -   Synchronized
DEV055   0832 RW       0       0 RW 0832 WD       0       0 C.D.  .  -   Synchronized
DEV056   0833 RW       0       0 RW 0833 WD       0       0 C.D.  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0    8192                  0       0
  MBs                0.0   512.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:
```

```
 Cons(istency State)    : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)        : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following `symcg show` command displays periods (.) under **C** in **`Flags,`** indicating the Consistency state of the devices is currently disabled:

```
‹R1Host#› symcg show SRDF

Composite Group Name:  SRDF

  Composite Group Type                                : RDF1
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : Yes
  RDF Consistency Protection Allowed                  : Yes
  RDF Consistency Mode                                : NONE
  Concurrent RDF                                      : No
  Cascaded RDF                                        : No

  Number of RDF (RA) Groups                           :    2
  Number of STD Devices                               :   56
  Number of CRDF STD Devices                          :    0
  Number of BCV's (Locally-associated)                :    0
  Number of VDEV's (Locally-associated)               :    0
  Number of TGT's Locally-associated                  :    0
  Number of CRDF TGT Devices                          :    0
  Number of RVDEV's (Remotely-associated VDEV)        :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)     :    0
  Number of RRBCV's (Remotely-associated RBCV)        :    0
  Number of RTGT's (Remotely-associated)              :    0
  Number of Hop2 BCV's (Remotely-assoc'ed Hop2 BCV)   :    0
  Number of Hop2 VDEV's (Remotely-assoc'ed Hop2 VDEV):    0
  Number of Hop2 TGT's (Remotely-assoc'ed Hop2 TGT)   :    0
  Number of Device Groups                             :    0
  Device Group Names                                  : N/A

  Number of Symmetrix Units (1):
    {

    1) Symmetrix ID                                    : 000192600321
       Microcode Version                               : 5875
       Number of STD Devices                           :   56
       Number of CRDF STD Devices                      :    0
       Number of BCV's (Locally-associated)            :    0
       Number of VDEV's (Locally-associated)           :    0
       Number of TGT's Locally-associated              :    0
       Number of CRDF TGT Devices                      :    0
       Number of RVDEV's (Remotely-associated VDEV)    :    0
       Number of RBCV's (Remotely-associated STD_RDF)  :    0
       Number of BRBCV's (Remotely-associated BCV-RDF):    0
       Number of RTGT's (Remotely-associated)          :    0
       Number of RRBCV's (Remotely-associated RBCV)    :    0
       Number of Hop2BCV's (Remotely-assoc'ed Hop2BCV):    0
       Number of Hop2VDEVs(Remotely-assoc'ed Hop2VDEV):    0
       Number of Hop2TGT's (Remotely-assoc'ed Hop2TGT):    0

       Number of RDF (RA) Groups (2):
         {

         1) RDF (RA) Group Number  : 128            (7F)
            Remote Symmetrix ID    : 000192600256
            Microcode Version      : 5875
            Recovery RA Group      : N/A            (N/A)
            RA Group Name          : N/A
```

```
STD Devices (28):
  {
  ------------------------------------------------------------------------------
                                          Sym  Device              Flags      Cap
  LdevName    PdevName                    Dev  Config        Sts   CSRT      (MB)
  ------------------------------------------------------------------------------
  DEV001      /dev/rhdisk27               0060 RDF1+Mir      RW    .--1      4314
  DEV002      /dev/rhdisk28               0061 RDF1+Mir      RW    .--1      4314
  DEV003      /dev/rhdisk29               0062 RDF1+Mir      RW    .--1      4314
  DEV004      /dev/rhdisk30               0063 RDF1+Mir      RW    .--1      4314
  DEV005      /dev/rhdisk31               0064 RDF1+Mir      RW    .--1      4314
  DEV006      /dev/rhdisk32               0065 RDF1+Mir      RW    .--1      4314
  DEV007      /dev/rhdisk33               0066 RDF1+Mir      RW    .--1      4314
  DEV008      /dev/rhdisk34               0067 RDF1+Mir      RW    .--1      4314
  DEV009      /dev/rhdisk43               07E0 RDF1+TDEV     RW    .--1      4314
  DEV010      /dev/rhdisk44               07E1 RDF1+TDEV     RW    .--1      4314
  DEV011      /dev/rhdisk45               07E2 RDF1+TDEV     RW    .--1      4314
  DEV012      /dev/rhdisk46               07E3 RDF1+TDEV     RW    .--1      4314
  DEV013      /dev/rhdisk47               07E4 RDF1+TDEV     RW    .--1      4314
  DEV014      /dev/rhdisk48               07E5 RDF1+TDEV     RW    .--1      4314
  DEV015      /dev/rhdisk49               07E6 RDF1+TDEV     RW    .--1      4314
  DEV016      /dev/rhdisk50               07E7 RDF1+TDEV     RW    .--1      4314
  DEV017      /dev/rhdisk51               07E8 RDF1+TDEV     RW    .--1      4314
  DEV018      /dev/rhdisk52               07E9 RDF1+TDEV     RW    .--1      4314
  DEV019      /dev/rhdisk53               07EA RDF1+TDEV     RW    .--1      4314
  DEV020      /dev/rhdisk54               07EB RDF1+TDEV     RW    .--1      4314
  DEV021      /dev/rhdisk55               07EC RDF1+TDEV     RW    .--1      4314
  DEV022      /dev/rhdisk56               07ED RDF1+TDEV     RW    .--1      4314
  DEV023      /dev/rhdisk57               07EE RDF1+TDEV     RW    .--1      4314
  DEV024      /dev/rhdisk58               07EF RDF1+TDEV     RW    .--1      4314
  DEV025      /dev/rhdisk76               0800 RDF1+TDEV     RW    .--1      4314
  DEV026      /dev/rhdisk216              0801 RDF1+TDEV     RW    .--1      4314
  DEV027      /dev/rhdisk217              0802 RDF1+TDEV     RW    .--1      4314
  DEV028      /dev/rhdisk218              0803 RDF1+TDEV     RW    .--1      4314
  }

2) RDF (RA) Group Number  : 129            (80)
   Remote Symmetrix ID    : 000192600256
   Microcode Version      : 5875
   Recovery RA Group      : N/A            (N/A)
   RA Group Name          : N/A

   STD Devices (28):
     {
     ------------------------------------------------------------------------------
                                          Sym  Device              Flags      Cap
     LdevName    PdevName                 Dev  Config        Sts   CSRT      (MB)
     ------------------------------------------------------------------------------
     DEV029      /dev/rhdisk223           0808 RDF1+TDEV     RW    .--1      4314
     DEV030      /dev/rhdisk224           0809 RDF1+TDEV     RW    .--1      4314
     DEV031      /dev/rhdisk225           080A RDF1+TDEV     RW    .--1      4314
     DEV032      /dev/rhdisk226           080B RDF1+TDEV     RW    .--1      4314
     DEV033      /dev/rhdisk227           080C RDF1+TDEV     RW    .--1      4314
     DEV034      /dev/rhdisk228           080D RDF1+TDEV     RW    .--1      4314
     DEV035      /dev/rhdisk229           080E RDF1+TDEV     RW    .--1      4314
     DEV036      /dev/rhdisk230           080F RDF1+TDEV     RW    .--1      4314
     DEV037      /dev/rhdisk231           0810 RDF1+TDEV     RW    .--1      4314
     DEV038      /dev/rhdisk232           0811 RDF1+TDEV     RW    .--1      4314
     DEV039      /dev/rhdisk233           0812 RDF1+TDEV     RW    .--1      4314
     DEV040      /dev/rhdisk234           0813 RDF1+TDEV     RW    .--1      4314
     DEV041      /dev/rhdisk235           0814 RDF1+TDEV     RW    .--1      4314
     DEV042      /dev/rhdisk236           0815 RDF1+TDEV     RW    .--1      4314
     DEV043      /dev/rhdisk237           0816 RDF1+TDEV     RW    .--1      4314
     DEV044      /dev/rhdisk238           0817 RDF1+TDEV     RW    .--1      4314
     DEV045      /dev/rhdisk254           0828 RDF1+TDEV     RW    .--1      4314
     DEV046      /dev/rhdisk255           0829 RDF1+TDEV     RW    .--1      4314
```

```
            DEV047      /dev/rhdisk256            082A RDF1+TDEV     RW    .--1     4314
            DEV048      /dev/rhdisk257            082B RDF1+TDEV     RW    .--1     4314
            DEV049      /dev/rhdisk258            082C RDF1+TDEV     RW    .--1     4314
            DEV050      /dev/rhdisk259            082D RDF1+TDEV     RW    .--1     4314
            DEV051      /dev/rhdisk260            082E RDF1+TDEV     RW    .--1     4314
            DEV052      /dev/rhdisk261            082F RDF1+TDEV     RW    .--1     4314
            DEV053      /dev/rhdisk262            0830 RDF1+TDEV     RW    .--1     4314
            DEV054      /dev/rhdisk263            0831 RDF1+TDEV     RW    .--1     4314
            DEV055      /dev/rhdisk264            0832 RDF1+TDEV     RW    .--1     4314
            DEV056      /dev/rhdisk265            0833 RDF1+TDEV     RW    .--1     4314
            }
        }
    }

Legend:
  RDFA Flags:
       C(onsistency)   : X = Enabled, . = Disabled, - = N/A
       (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
       R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
       (Mirror) T(ype) : 1 = R1, 2 = R2, - = N/A
```

- The following `symrdf set mode` command sets the mode to synchronous for the `SRDF` composite group:

  ‹R1Host#› **symrdf -cg SRDF set mode sync -noprompt**

  ```
  An RDF Set 'Synchronous Mode' operation execution is in
  progress for composite group 'SRDF'. Please wait...

  The RDF Set 'Synchronous Mode' operation successfully executed
  for composite group 'SRDF'.
  ```

- The following `symcg enable` command enables consistency for device pairs in the `SRDF` composite group:

  ```
  symcg -cg SRDF enable -noprompt
  ```

  ‹R1Host#› **symcg -cg SRDF enable -noprompt**

  ```
  A consistency 'Enable' operation execution is
  in progress for composite group 'SRDF'. Please wait...

  The consistency 'Enable' operation successfully executed for
  composite group 'SRDF'.
  ```

- The following query displays all pairs in the Synchronized state and an X in the C column under Modes to indicate that all pairs are now enabled for SRDF consistency:

‹R1Host#› **symrdf -cg SRDF query -detail**

```
Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  SYNC
Sync Consistency Info:

    Consistency State                  : Synchronized


Symmetrix ID                       : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 128 (7F)

        Source (R1) View             Target (R2) View       MODES
```

```
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   0060 RW       0       0 RW 0060 WD       0       0 S..X. Synchronized
DEV002   0061 RW       0       0 RW 0061 WD       0       0 S..X. Synchronized
DEV003   0062 RW       0       0 RW 0062 WD       0       0 S..X. Synchronized
DEV004   0063 RW       0       0 RW 0063 WD       0       0 S..X. Synchronized
DEV005   0064 RW       0       0 RW 0064 WD       0       0 S..X. Synchronized
DEV006   0065 RW       0       0 RW 0065 WD       0       0 S..X. Synchronized
DEV007   0066 RW       0       0 RW 0066 WD       0       0 S..X. Synchronized
DEV008   0067 RW       0       0 RW 0067 WD       0       0 S..X. Synchronized
DEV009   07E0 RW       0       0 RW 07E0 WD       0       0 S..X. Synchronized
DEV010   07E1 RW       0       0 RW 07E1 WD       0       0 S..X. Synchronized
DEV011   07E2 RW       0       0 RW 07E2 WD       0       0 S..X. Synchronized
DEV012   07E3 RW       0       0 RW 07E3 WD       0       0 S..X. Synchronized
DEV013   07E4 RW       0       0 RW 07E4 WD       0       0 S..X. Synchronized
DEV014   07E5 RW       0       0 RW 07E5 WD       0       0 S..X. Synchronized
DEV015   07E6 RW       0       0 RW 07E6 WD       0       0 S..X. Synchronized
DEV016   07E7 RW       0       0 RW 07E7 WD       0       0 S..X. Synchronized
DEV017   07E8 RW       0       0 RW 07E8 WD       0       0 S..X. Synchronized
DEV018   07E9 RW       0       0 RW 07E9 WD       0       0 S..X. Synchronized
DEV019   07EA RW       0       0 RW 07EA WD       0       0 S..X. Synchronized
DEV020   07EB RW       0       0 RW 07EB WD       0       0 S..X. Synchronized
DEV021   07EC RW       0       0 RW 07EC WD       0       0 S..X. Synchronized
DEV022   07ED RW       0       0 RW 07ED WD       0       0 S..X. Synchronized
DEV023   07EE RW       0       0 RW 07EE WD       0       0 S..X. Synchronized
DEV024   07EF RW       0       0 RW 07EF WD       0       0 S..X. Synchronized
DEV025   0800 RW       0       0 RW 0800 WD       0       0 S..X. Synchronized
DEV026   0801 RW       0       0 RW 0801 WD       0       0 S..X. Synchronized
DEV027   0802 RW       0       0 RW 0802 WD       0       0 S..X. Synchronized
DEV028   0803 RW       0       0 RW 0803 WD       0       0 S..X. Synchronized

Symmetrix ID                      : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number             : 129 (80)

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV029   0808 RW       0       0 RW 0808 WD       0       0 S..X. Synchronized
DEV030   0809 RW       0       0 RW 0809 WD       0       0 S..X. Synchronized
DEV031   080A RW       0       0 RW 080A WD       0       0 S..X. Synchronized
DEV032   080B RW       0       0 RW 080B WD       0       0 S..X. Synchronized
DEV033   080C RW       0       0 RW 080C WD       0       0 S..X. Synchronized
DEV034   080D RW       0       0 RW 080D WD       0       0 S..X. Synchronized
DEV035   080E RW       0       0 RW 080E WD       0       0 S..X. Synchronized
DEV036   080F RW       0       0 RW 080F WD       0       0 S..X. Synchronized
DEV037   0810 RW       0       0 RW 0810 WD       0       0 S..X. Synchronized
DEV038   0811 RW       0       0 RW 0811 WD       0       0 S..X. Synchronized
DEV039   0812 RW       0       0 RW 0812 WD       0       0 S..X. Synchronized
DEV040   0813 RW       0       0 RW 0813 WD       0       0 S..X. Synchronized
DEV041   0814 RW       0       0 RW 0814 WD       0       0 S..X. Synchronized
DEV042   0815 RW       0       0 RW 0815 WD       0       0 S..X. Synchronized
DEV043   0816 RW       0       0 RW 0816 WD       0       0 S..X. Synchronized
DEV044   0817 RW       0       0 RW 0817 WD       0       0 S..X. Synchronized
DEV045   0828 RW       0       0 RW 0828 WD       0       0 S..X. Synchronized
DEV046   0829 RW       0       0 RW 0829 WD       0       0 S..X. Synchronized
DEV047   082A RW       0       0 RW 082A WD       0       0 S..X. Synchronized
DEV048   082B RW       0       0 RW 082B WD       0       0 S..X. Synchronized
```

```
DEV049   082C RW       0       0 RW 082C WD       0       0 S..X. Synchronized
DEV050   082D RW       0       0 RW 082D WD       0       0 S..X. Synchronized
DEV051   082E RW       0       0 RW 082E WD       0       0 S..X. Synchronized
DEV052   082F RW       0       0 RW 082F WD       0       0 S..X. Synchronized
DEV053   0830 RW       0       0 RW 0830 WD       0       0 S..X. Synchronized
DEV054   0831 RW       0       0 RW 0831 WD       0       0 S..X. Synchronized
DEV055   0832 RW       0       0 RW 0832 WD       0       0 S..X. Synchronized
DEV056   0833 RW       0       0 RW 0833 WD       0       0 S..X. Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf suspend` command attempts to suspend the SRDF pairs in the `SRDF` composite group. The `-force` option ensures that you want to stop the SRDF mirroring operation and end consistency protection for the composite group.

‹R1Host#› **symrdf -cg SRDF suspend -noprompt**

```
An RDF 'Suspend' operation execution is
in progress for composite group 'SRDF'. Please wait...

Cannot proceed in the current RDF Consistency state except if the force flag is used
```

- The following `symrdf suspend` command with the `-force` option successfully suspends the SRDF pairs in the `SRDF` composite group:

‹R1Host#› **symrdf -cg SRDF suspend -force -noprompt**

```
An RDF 'Suspend' operation execution is
in progress for composite group 'SRDF'. Please wait...

    Pend I/O on RDF link(s) for device(s) in (0321,128)............Done.
    Pend I/O on RDF link(s) for device(s) in (0321,129)............Done.
    Suspend RDF link(s) for device(s) in (0321,128)................Done.

The RDF 'Suspend' operation successfully executed for
composite group 'SRDF'.
```

Example 7: Creating concurrent dynamic SRDF pairs

In this example, the host is connected to the local Symmetrix array (sid 321) that is connected to two remote Symmetrix arrays (sids 198 and 256) using two different SRDF (RA) groups for connectivity. This example assumes that dynamic SRDF is configured for both the local and remote Symmetrix arrays, and the three Symmetrix arrays are enabled for a concurrent SRDF relationship.

- The `symcfg list` command displays the SRDF groups linking the local Symmetrix array to the remote Symmetrix arrays through director `7f`:

‹R1Host#› **symcfg list -ra 7f -sid 321**

```
Symmetrix ID: 000192600321
```

```
                    S Y M M E T R I X   R D F   D I R E C T O R S

                                                Remote        Local    Remote
Ident  Symb  Num  Slot  Type       Attr  SymmID        RA Grp   RA Grp  Status

RF-7F   07F   87     7  RDF-R2       -   000192600256 128 (7F) 128 (7F) Online
                                       -   000192600198 228 (E3) 228 (E3)
                                       -   000192600256   2 (01)   2 (01)
                                       -   000192600256   2 (01)   2 (01)
                                       -   000192600256 128 (7F) 128 (7F)
                                       -   000192600256  35 (22)  35 (22)
                                       -   000192600198 111 (6E) 111 (6E)
                                       -   000192600198 228 (E3) 228 (E3)
```

- The following `symrdf addgrp` command creates a dynamic SRDF group representing another SRDF link between the local Symmetrix array and the other remote Symmetrix array (256). It adds the dynamic SRDF group 148 on the local Symmetrix array, and the SRDF group 148 on the remote Symmetrix array. You must specify a group label (in this case, `grp148`) to use when modifying or deleting this group. The creation of the dynamic SRDF group includes directors `7f,8f` from Symmetrix 321 and from Symmetrix 256 to use as connection points.

  When creating dynamic SRDF groups between two Symmetrix arrays, it is important to understand the network topology when choosing director endpoints. If using the Fibre Channel protocol, the director endpoints must see each other through the FC fabric to create the dynamic SRDF links. Ensure that the physical connections between the local RA and remote RA are valid and operational. Static and dynamic groups may co-exist on the same RA directors.

```
‹R1Host#› symrdf addgrp -label grp148 -sid 321 -rdfg 148 -dir 7f,8f \
   -remote_sid 256 -remote_rdfg 148 -remote_dir 7f,8f

 Successfully Added Dynamic RDF Group 'grp148' for Symm: 000192600321
```

- The following `symcfg list` command shows the remote connection of the SRDF group 148 from the local Symmetrix array. In this display, the SRDF group 148 connects Symmetrix 321 to Symmetrix 256.

```
‹R1Host#› symcfg list -sid 321 -rdfg 148

Symmetrix ID : 000192600321

               S Y M M E T R I X   R D F   G R O U P S

    Local             Remote                  Group                RDFA Info
-------------- --------------------- -------------------------- ---------------
            LL                                   Flags   Dir Flags Cycle
 RA-Grp  (sec)  RA-Grp  SymmID       T    Name    LPDS CHT  Cfg CSRM  time  Pri
-------------- --------------------- -------------------------- ----- ----- ---
148 (93)    10 148 (93) 000192600256 D grp148     XX.. ..X  F-S -IS-     30  33

Legend:
  ?                 : Unknown
  Group (T)ype      :  S = Static, D = Dynamic
  Director (C)onfig :  F-S = Fibre-Switched, F-H = Fibre-Hub
                       G = GIGE, E = ESCON, T = T3, - = N/A
  Group Flags       :
      Prevent Auto (L)ink Recovery      : X = Enabled, . = Disabled
      Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
      Link (D)omino                     : X = Enabled, . = Disabled
```

```
    (S)TAR mode                       : N = Normal, R = Recovery, . = OFF
    RDF Software (C)ompression        : X = Enabled, . = Disabled, - = N/A
    RDF (H)ardware Compression        : X = Enabled, . = Disabled, - = N/A
    RDF Single Round (T)rip           : X = Enabled, . = Disabled, - = N/A
RDFA Flags        :
    (C)onsistency : X = Enabled, . = Disabled, - = N/A
    (S)tatus      : A = Active, I = Inactive, - = N/A
    (R)DFA Mode   : S = Single-session, M = MSC, - = N/A
    (M)sc Cleanup : C = MSC Cleanup required, - = N/A
```

- The following `symrdf addgrp` command creates a dynamic SRDF group that represents another link between Symmetrix 321 and Symmetrix 198. It adds dynamic SRDF group 149 on the local Symmetrix array, and SRDF group 149 on the remote Symmetrix array (198). The creation of the dynamic SRDF group includes two sets of connection endpoints (director `7f,8f`) from Symmetrix 321 and Symmetrix 198.

```
‹R1Host#› symrdf addgrp -label grp149 -sid 321 -rdfg 149 -dir 7f,8f \
   -remote_sid 198 -remote_rdfg 149 -remote_dir 7f,8f

 Successfully Added Dynamic RDF Group 'grp149' for Symm: 000192600321
```

- The following `symcfg list` command shows the remote connection of the SRDF group 149 from the local Symmetrix array. In this display, the SRDF group 149 connects Symmetrix 321 to Symmetrix 198.

```
‹R1Host#› symcfg list -sid 321 -rdfg 149

Symmetrix ID : 000192600321

                S Y M M E T R I X   R D F   G R O U P S

    Local             Remote                   Group                 RDFA Info
-------------- --------------------- --------------------------- ---------------
             LL                                     Flags   Dir Flags Cycle
 RA-Grp   (sec)  RA-Grp  SymmID       T    Name    LPDS CHT  Cfg CSRM  time  Pri
-------------- --------------------- --------------------------- ----- ----- ---
149 (94)     10 149 (94) 000192600198 D grp149     XX.. ..X  F-S -IS-     30  33
```

- The following `symdev list` command with the `-dynamic` option displays the source devices configured for SRDF dynamic capability on Symmetrix 321:

```
‹R1Host#› symdev list -sid 321 -dynamic -r1 -r2

Symmetrix ID: 000192600321

        Device Name           Directors                   Device
--------------------------- ------------- -------------------------------------
                                                                          Cap
Sym  Physical               SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- -------------------------------------

0060 /dev/rhdisk27          10E:0 10A:D0  2-Way Mir     N/Grp'd      RW    4314
0061 /dev/rhdisk28          10E:0 09C:C0  2-Way Mir     N/Grp'd      RW    4314
0062 /dev/rhdisk29          10E:0 10B:C0  2-Way Mir     N/Grp'd      RW    4314
0063 /dev/rhdisk30          10E:0 09D:D0  2-Way Mir     N/Grp'd      RW    4314
0064 /dev/rhdisk31          10E:0 09A:C2  2-Way Mir     N/Grp'd      RW    4314
0065 /dev/rhdisk32          10E:0 10C:D2  2-Way Mir     N/Grp'd      RW    4314
0066 /dev/rhdisk33          10E:0 09B:D2  2-Way Mir     N/Grp'd      RW    4314
.
.
.
.
07E8 /dev/rhdisk51          10E:0  NA:NA  TDEV          N/Grp'd      RW    4314
07E9 /dev/rhdisk52          10E:0  NA:NA  TDEV          N/Grp'd      RW    4314
```

```
07EA /dev/rhdisk53          10E:0  NA:NA  TDEV          N/Grp'd      RW    4314
```

- The following command uses the vi text editor to create the `RDFG148` file consisting of dynamic SRDF pairs (R1/R2) for the local Symmetrix 321 and the remote Symmetrix 256. The R1 devices are in the first column with their corresponding remote R2 devices in the second column. For example, the first line shows the pairing of 0060 (R1) with 0060 (R2).

  ```
  <R1Host#> vi RDFG148
  0060 0060
  0061 0061
  ```

- The following `symrdf createpair` command establishes the dynamic pairs by parsing the above `RDFG148` file and specifying its column-1 devices as source devices on Symmetrix 321. Because this command is using SRDF group 148, communication is between Symmetrix 321 and remote Symmetrix 256.

```
<R1Host#> symrdf createpair -sid 321 -rdfg 148 -file RDFG148 -type rdf1 -establish
   -noprompt

An RDF 'Create Pair' operation execution is in progress for device
file 'RDFG148'. Please wait...

    Create RDF Pair in (0321,148)....................................Started.
    Create RDF Pair in (0321,148)....................................Done.
    Mark target device(s) in (0321,148) for full copy from source....Started.
    Devices: 0060-0061 in (0321,148)............................... Marked.
    Mark target device(s) in (0321,148) for full copy from source....Done.
    Merge track tables between source and target in (0321,148).......Started.
    Devices: 0060-0061 in (0321,148)............................... Merged.
    Merge track tables between source and target in (0321,148).......Done.
    Resume RDF link(s) for device(s) in (0321,148)...................Started.
    Resume RDF link(s) for device(s) in (0321,148)...................Done.

The RDF 'Create Pair' operation successfully executed for device
file 'RDFG148'.
```

- The following command creates the `RDFG149` file consisting of dynamic SRDF pairs (R1/R2) for the local Symmetrix 321 and the remote Symmetrix 198. The R1 devices are in the first column with their corresponding remote R2 devices in the second column.

  ```
  <R1Host#> vi RDFG149
  0060 0054
  0061 0055
  ```

- The following `symrdf createpair` command establishes the dynamic pairs by parsing the above `RDFG149` file and specifying its column-1 devices as source devices on Symmetrix 321. Because this command is using SRDF group 149, communication is between Symmetrix 321 and remote Symmetrix 198.

```
<R1Host#> symrdf createpair -sid 321 -rdfg 149 -file RDFG149 -type rdf1 -establish
   -noprompt

An RDF 'Create Pair' operation execution is in progress for device
file 'RDFG149'. Please wait...

    Create RDF Pair in (0321,149)....................................Started.
    Create RDF Pair in (0321,149)....................................Done.
    Mark target device(s) in (0321,149) for full copy from source....Started.
    Devices: 0060-0061 in (0321,149)............................... Marked.
    Mark target device(s) in (0321,149) for full copy from source....Done.
    Merge track tables between source and target in (0321,149).......Started.
```

```
    Devices: 0060-0061 in (0321,149)............................... Merged.
    Merge track tables between source and target in (0321,149).......Done.
    Resume RDF link(s) for device(s) in (0321,149)..................Started.
    Resume RDF link(s) for device(s) in (0321,149)..................Done.

The RDF 'Create Pair' operation successfully executed for device
file 'RDFG149'.
```

- The `symdg` command creates an RDF1 type device group named `dynConc`. The `symdg` command adds the R1 devices belonging to SRDF group 148 on Symmetrix 321 to `dynConc`.

  ```
  ‹R1Host#› symdg create dynConc -type rdf1
  ‹R1Host#› symdg -g dynConc -sid 321 addall dev -sel_rdfg 148 -v

  Device(s) 00060:00061: ADD DEVS BY RANGE Succeeded.
  ```

- The `symrdf query –rdfg all` command displays the concurrent SRDF pairings for the local R1 devices in the device group `dynConc`. The `–rdfg` option allows you to see the SRDF pairs represented by both SRDF (RA) groups. As shown, all the concurrent pairs are in the Synchronized state.

```
‹R1Host#› symrdf -g dynConc query -rdfg all

Device Group (DG) Name             : dynConc
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600321    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 148 (93)
Remote Symmetrix ID                : 000192600198    (Microcode Version: 5875)
RDF (RA) Group Number              : 149 (94)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 RW       0        0 RW 0060 WD       0        0 S...  Synchronized
             RW       0        0 RW 0054 WD       0        0 S...  Synchronized
DEV002  0061 RW       0        0 RW 0061 WD       0        0 S...  Synchronized
             RW       0        0 RW 0055 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

Example 8: Failing over cascaded SRDF

This example shows how to perform a planned failover of a cascaded SRDF configuration using the sites shown in Figure 60. All the SYMCLI commands in this example are run from the host attached to SiteC. In this configuration, the primary site, SiteA, is two hops away from where the commands are executed.

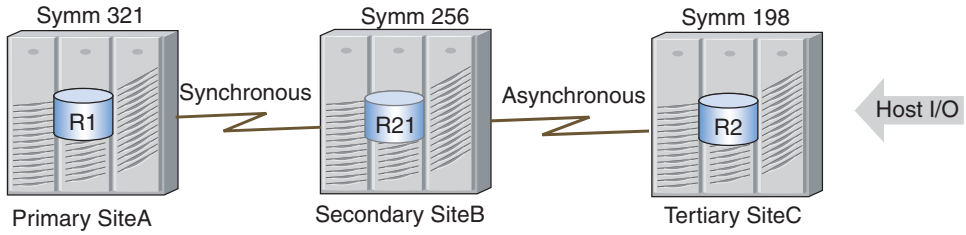


Figure 60 Initial cascaded SRDF configuration

- The following `symcfg list` command displays the Symmetrix arrays visible to the host attached to SiteC:

```
‹R2host#› symcfg list

                                S Y M M E T R I X

                                          Mcode     Cache      Num Phys  Num Symm
    SymmID        Attachment  Model       Version   Size (MB)  Devices   Devices

    000192600198  Local       VMAX-1      5875        24576       145      2953
    000192600256  Remote      VMAX-1      5875        24576         0      2953
    000192600321  Remote      VMAX-1      5875        24576         0      2953
```

- The following query shows that SiteA to SiteB devices are in a Synchronized state. The SiteB to SiteC devices are in a Consistent SRDF pair state, as shown in the next display.

```
‹R2Host#› symrdf -g DGrp_CARDF query -hop2

Device Group (DG) Name              : DGrp_CARDF
DG's Type                           : RDF2
DG's Symmetrix ID                   : 000192600198    (Microcode Version: 5875)
Hop-2 Symmetrix ID                  : 000192600256    (Microcode Version: 5875)
Hop-2 Remote Symmetrix ID           : 000192600321    (Microcode Version: 5875)
DG RDF (RA) Group Number            : 178 (B1)
Hop-2 RDF (RA) Group Number         : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 RW 0060 RW       0        0 S...  Synchronized
DEV002  0061 WD       0        0 RW 0061 RW       0        0 S...  Synchronized
DEV003  0062 WD       0        0 RW 0062 RW       0        0 S...  Synchronized
.
.
.
```

```
.
DEV026  0801 WD       0        0 RW 0801 RW       0        0 S...  Synchronized
DEV027  0802 WD       0        0 RW 0802 RW       0        0 S...  Synchronized
DEV028  0803 WD       0        0 RW 0803 RW       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following query shows that the SiteB to SiteC devices are in a Consistent SRDF pair state:

```
‹R2Host#› symrdf -g DGrp_CARDF query -rdfa

Device Group (DG) Name             : DGrp_CARDF
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600198    (Microcode Version: 5875)

RDFA Session Number                : 177
RDFA Cycle Number                  : 4
RDFA Session Status                : Active
RDFA Consistency Exempt Devices    : No
RDFA Minimum Cycle Time            : 00:00:30
RDFA Avg Cycle Time                : 00:00:45
Duration of Last cycle             : 00:00:30
RDFA Session Priority              : 33
Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1          : 00:00:55
R2 Image Capture Time              : Fri Jul 23 17:41:52 2010
R2 Data is Consistent              : True
RDFA R1 Side Percent Cache In Use  :  0
RDFA R2 Side Percent Cache In Use  :  0
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
Transmit Idle Time                 : 00:00:00
           R1 Side Shared Tracks              : 0

Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 178 (B1)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDACE STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 RW 0060 RW       0        0 A..X. Consistent
DEV002  0061 WD       0        0 RW 0061 RW       0        0 A..X. Consistent
DEV003  0062 WD       0        0 RW 0062 RW       0        0 A..X. Consistent
.
.
.
.
DEV026  0801 WD       0        0 RW 0801 RW       0        0 A..X. Consistent
DEV027  0802 WD       0        0 RW 0802 RW       0        0 A..X. Consistent
DEV028  0803 WD       0        0 RW 0803 RW       0        0 A..X. Consistent

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following command first suspends the device pairs of the SiteA to SiteB relationship using device group `DGrp_CARDF`. Since SiteA is two hops away from SiteC, you must use the `-hop2` option.

```
‹R2Host#› symrdf -g DGrp_CARDF suspend -hop2

An RDF 'Suspend' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Suspend RDF link(s).....................................Done.

The RDF 'Suspend' operation successfully executed for
device group 'DGrp_CARDF'.
```

- The following query verifies that the SiteA to SiteB devices are now in a Suspended SRDF pair state:

```
‹R2Host#› symrdf -g DGrp_CARDF query -hop2

Device Group (DG) Name             : DGrp_CARDF
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600198    (Microcode Version: 5875)
Hop-2 Symmetrix ID                 : 000192600256    (Microcode Version: 5875)
Hop-2 Remote Symmetrix ID          : 000192600321    (Microcode Version: 5875)
DG RDF (RA) Group Number           : 178 (B1)
Hop-2 RDF (RA) Group Number        : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 NR 0060 RW       0        0 S...  Suspended
DEV002  0061 WD       0        0 NR 0061 RW       0        0 S...  Suspended
DEV003  0062 WD       0        0 NR 0062 RW       0        0 S...  Suspended
.
.
.
.
DEV026  0801 WD       0        0 NR 0801 RW       0        0 S...  Suspended
DEV027  0802 WD       0        0 NR 0802 RW       0        0 S...  Suspended
DEV028  0803 WD       0        0 NR 0803 RW       0        0 S...  Suspended
```

- The following command suspends the device pairs of the SiteB to SiteC relationship using `DGrp_CARDF`. Note that you must use the `-force` option:

```
‹R2Host#› symrdf -g DGrp_CARDF suspend

An RDF 'Suspend' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

Cannot proceed in the current RDF Consistency state except if the force flag is used

‹R2Host#› symrdf -g DGrp_CARDF suspend -force

An RDF 'Suspend' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Suspend RDF link(s).....................................Started.
    Suspend RDF link(s).....................................Done.
```

```
The RDF 'Suspend' operation successfully executed for
device group 'DGrp_CARDF'.
```

- The following query verifies that the SiteB to SiteC devices are now in a Suspended SRDF pair state:

```
‹R2Host#› symrdf -g DGrp_CARDF query

Device Group (DG) Name             : DGrp_CARDF
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600198    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 178 (B1)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 NR 0060 RW       0        0 A...  Suspended
DEV002  0061 WD       0        0 NR 0061 RW       0        0 A...  Suspended
DEV003  0062 WD       0        0 NR 0062 RW       0        0 A...  Suspended
.
.
.
.
DEV026  0801 WD       0        0 NR 0801 RW       0        0 A...  Suspended
DEV027  0802 WD       0        0 NR 0802 RW       0        0 A...  Suspended
DEV028  0803 WD       0        0 NR 0803 RW       0        0 A...  Suspended

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following command performs the failover operation for SiteA to SiteB using `DGrp_CARDF`:

```
‹R2Host#› symrdf -g DGrp_CARDF failover -hop2

An RDF 'Failover' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Write Disable device(s) on SA at source (R1)..............Done.
    Suspend RDF link(s)......................................Done.
    Read/Write Enable device(s) on RA at target (R2)..........Done.

The RDF 'Failover' operation successfully executed for
device group 'DGrp_CARDF'.
```

- The following command performs the failover operation for SiteB to SiteC using `DGrp_CARDF`. Note that you must use the `-force` option.

```
‹R2Host#› symrdf -g DGrp_CARDF failover

An RDF 'Failover' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

Cannot proceed in the current RDF Consistency state except if the force flag is used

‹R2Host#› symrdf -g DGrp_CARDF failover -force
```

```
An RDF 'Failover' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Write Disable device(s) on SA at source (R1).............Done.
    Suspend RDF link(s)......................................Done.
    Suspend RDF link(s)......................................Started.
    Suspend RDF link(s)......................................Done.
    Read/Write Enable device(s) on RA at target (R2)..........Done.
    Suspend RDF link(s)......................................Started.
    Suspend RDF link(s)......................................Done.
```

- The following query confirms that the pair states of SiteA to SiteB have failed over:

‹R2Host#› **symrdf -g DGrp_CARDF query -hop2**

```
Device Group (DG) Name             : DGrp_CARDF
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600198    (Microcode Version: 5875)
Hop-2 Symmetrix ID                 : 000192600256    (Microcode Version: 5875)
Hop-2 Remote Symmetrix ID          : 000192600321    (Microcode Version: 5875)
DG RDF (RA) Group Number           : 178 (B1)
Hop-2 RDF (RA) Group Number        : 128 (7F)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0060 WD       0        0 NR 0060 WD       0        0 S...  Failed Over
DEV002  0061 WD       0        0 NR 0061 WD       0        0 S...  Failed Over
DEV003  0062 WD       0        0 NR 0062 WD       0        0 S...  Failed Over
.
.
.
.
.
DEV026  0801 WD       0        0 NR 0801 WD       0        0 S...  Failed Over
DEV027  0802 WD       0        0 NR 0802 WD       0        0 S...  Failed Over
DEV028  0803 WD       0        0 NR 0803 WD       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following query confirms that the pair states of SiteB to SiteC have failed over:

‹R2Host#› **symrdf -g DGrp_CARDF query**

```
Device Group (DG) Name             : DGrp_CARDF
DG's Type                          : RDF2
DG's Symmetrix ID                  : 000192600198    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192600256    (Microcode Version: 5875)
RDF (RA) Group Number              : 178 (B1)


       Target (R2) View                 Source (R1) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------
```

```
DEV001  0060 RW       0         0 NR 0060 WD       0         0 A...  Failed Over
DEV002  0061 RW       0         0 NR 0061 WD       0         0 A...  Failed Over
DEV003  0062 RW       0         0 NR 0062 WD       0         0 A...  Failed Over
.
.
.
.
DEV026  0801 RW       0         0 NR 0801 WD       0         0 A...  Failed Over
DEV027  0802 RW       0         0 NR 0802 WD       0         0 A...  Failed Over
DEV028  0803 RW       0         0 NR 0803 WD       0         0 A...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0
```

- The following command sets the SRDF mode to adaptive copy disk for the link between SiteA and SiteB, as shown in Figure 61:

```
‹R2Host#› symrdf -g DGrp_CARDF set mode acp_disk -hop2

An RDF Set 'ACp Disk Mode ON' operation execution is in
progress for device group 'DGrp_CARDF'. Please wait...

The RDF Set 'ACp Disk Mode ON' operation successfully executed
for device group 'DGrp_CARDF'.
```

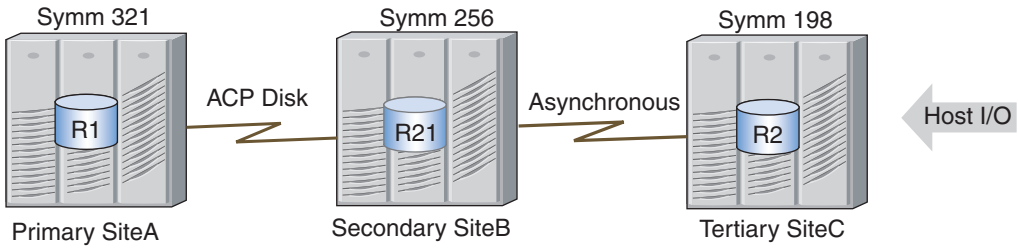


Figure 61 SRDF mode is set to ACP Disk between SiteA and SiteB

- The following command swaps the personality of the SiteA and SiteB devices using `DGrp_CARDF`. SiteB becomes the concurrent R1 (R11) site and SiteA becomes the R2 site, as shown in Figure 62.

```
‹R2Host#› symrdf -g DGrp_CARDF swap -hop2

An RDF 'Swap Personality' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Swap RDF Personality.....................................Started.
    Swap RDF Personality.....................................Done.

The RDF 'Swap Personality' operation successfully executed for
```

```
device group 'DGrp_CARDF'.
```

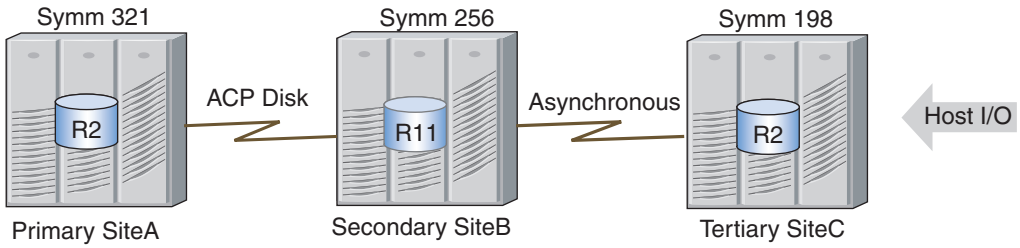


Figure 62 Personality swap between SiteA and SiteB devices

- The following command swaps the personality of the SiteB and SiteC devices using `DGrp_CARDF`. SiteB becomes the R21 site and SiteC becomes the R1 site, as shown in Figure 63.

```
‹R2Host#› symrdf -g DGrp_CARDF swap

An RDF 'Swap Personality' operation execution is
in progress for device group 'DGrp_CARDF'. Please wait...

    Suspend RDF link(s).......................................Started.
    Suspend RDF link(s).......................................Done.
    Swap RDF Personality......................................Started.
    Swap RDF Personality......................................Done.

The RDF 'Swap Personality' operation successfully executed for
device group 'DGrp_CARDF'.
```

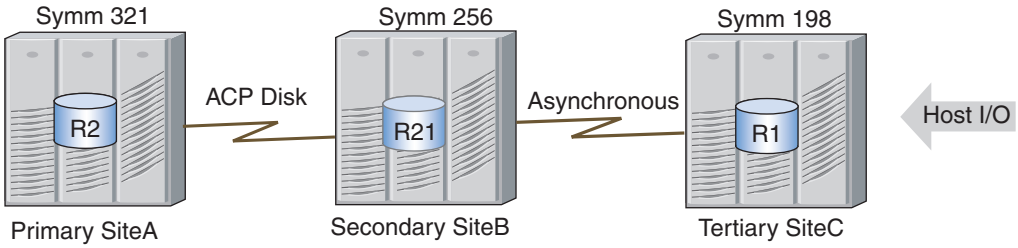


Figure 63 Personality swap between SiteB and SiteC devices

CHAPTER 14
Querying and Verifying with SRDF Commands

This chapter provides examples of the SYMCLI actions and specific commands, used to query and verify SRDF group operations. It focuses on the various arguments, options, and the application of certain parameters for the SRDF query and verify actions. Using examples of SRDF commands, it describes how to manage the behavior and states of the various SRDF components in a typical configuration. Topics include:

- Example 1: Querying a device group 350
- Example 2: Querying a composite group 367

Note: The commands in this chapter were executed using Solutions Enabler V7.3.

Example 1: Querying a device group

Before creating a device group and adding devices to it, examine the devices on your local Symmetrix array to determine which are source devices (Sym Dev), which are remote target devices (RDev), and whether a device is an R1 or R2 type device. The `symrdf list` command displays this information and other relevant data such as SRDF group (G), replication method (column M), pair state, invalid tracks, and the state of each device and the SRDF links that connect them.

```
symrdf list

Symmetrix ID: 000192600282

                              Local Device View
---------------------------------------------------------------------------
                     STATUS     MODES                       RDF  S T A T E S
Sym        RDF      ---------  -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G    SA RA LNK  MDATE  Tracks   Tracks Dev RDev Pair
---- ---- --------  --------- ----- -------  ------- --- ---- -------------

0060 0060   R1:98   RW RW NR   A..1.       0       69 RW  WD   Suspended
0061 0061   R1:98   RW RW NR   A..1.       0       69 RW  WD   Suspended
0068 0068   R1:98   RW RW NR   A..1X       0        0 RW  WD   Suspended
0069 0069   R1:98   RW RW NR   A..1X       0        0 RW  WD   Suspended
00F0 00F0   R2:118  RW WD RW   S..2.       0        0 WD  RW   Synchronized
00F1 00F1   R2:118  RW WD RW   S..2.       0        0 WD  RW   Synchronized
0108 0108   R1:108  RW RW NR   C.D1.       0        0 RW  WD   Suspended
0109 0109   R1:108  RW RW NR   C.D1.       0        0 RW  WD   Suspended
0110 0110   R1:97   RW RW NR   A..1.       0        0 RW  RW   Split
0111 0111   R1:97   RW RW NR   A..1.       0        0 RW  RW   Split
```

- The `symdev list` command with the `-r1` option displays all R1 devices. Those R1 devices that are not already part of a device group are displayed as N/Grp'd, which means they are available to be added to a new RDF1 device group:

```
symdev list -r1

Symmetrix ID: 000192600282

        Device Name                                    Directors                 Device
------------------------------------------------- ------------- ------------------------------------
                                                                                                Cap
Sym  Physical                                     SA :P DA :IT  Config        Attribute    Sts   (MB)
------------------------------------------------- ------------- ------------------------------------

0060 /dev/rdsk/c55t5d4                            08E:1 09A:D1  RDF1+Mir      Grp'd        RW    4314
0061 /dev/rdsk/c55t5d5                            08E:1 10C:C1  RDF1+Mir      Grp'd        RW    4314
0108 /dev/rdsk/c56t10d4                           08E:1 10B:D3  RDF1+Mir      N/Grp'd      RW    4314
0109 /dev/rdsk/c56t10d5                           08E:1 08D:D3  RDF1+Mir      N/Grp'd      RW    4314
```

- The following `symdg create` command creates the `Rdf1Grp` device group. The `symdg add` command adds standard devices to the group, using either a device's physical device (`pd`) name or, as shown below, its Symmetrix device (`dev`) name.

```
symdg create Rdf1Grp -type r1
symdg -g Rdf1Grp -sid 282 add dev 108
symdg -g Rdf1Grp -sid 282 add dev 109
symdg show Rdf1Grp

Group Name:  Rdf1Grp

    Group Type                                  : RDF1     (RDFA)
    Device Group in GNS                         : No
    Valid                                       : Yes
    Symmetrix ID                                : 000192600282
    Group Creation Time                         : Tue Apr 12 16:04:16 2011
    Vendor ID                                   : EMC Corp
    Application ID                              : SYMCLI
```

```
Number of STD Devices in Group               :    2
Number of Associated GK's                    :    0
Number of Locally-associated BCV's           :    0
Number of Locally-associated VDEV's          :    0
Number of Locally-associated TGT's           :    0
Number of Remotely-associated VDEV's(STD RDF):    0
Number of Remotely-associated BCV's (STD RDF):    0
Number of Remotely-associated TGT's(TGT RDF) :    0
Number of Remotely-associated BCV's (BCV RDF):    0
Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0
Number of Composite Groups                   :    0
Composite Group Names                        : N/A

Standard (STD) Devices (2):
    {
    ----------------------------------------------------------------------------------------------------
                                                                        Sym  Device                      Cap
    LdevName              PdevName                                      Dev  Config        Att. Sts      (MB)
                            ------------------------------------------------------------------------------
    DEV001                /dev/rdsk/c56t10d4                            0108 RDF1+Mir           RW      4314
    DEV002                /dev/rdsk/c56t10d5                            0109 RDF1+Mir           RW      4314
    }

Device Group RDF Information
    {
    RDF Type                              : R1
    RDF (RA) Group Number                 : 108               (6B)

    Remote Symmetrix ID                   : 000192603139

    R2 Device Is Larger Than The R1 Device : False

      Paired with a Diskless Device         : False
    Paired with a Concurrent Device       : False
    Paired with a Cascaded Device         : False
    Thick Thin Relationship               : False

    RDF Pair Configuration                : Normal
    RDF STAR Mode                         : False

    RDF Mode                              : Adaptive Copy
    RDF Adaptive Copy                     : Enabled: Disk Mode
    RDF Adaptive Copy Write Pending State : N/A
    RDF Adaptive Copy Skew (Tracks)       : 32767

    RDF Device Domino                     : Disabled

    RDF Link Configuration                : Fibre
    RDF Link Domino                       : Disabled
    Prevent Automatic RDF Link Recovery   : Enabled
    Prevent RAs Online Upon Power ON      : Enabled

    Device RDF Status                     : Ready           (RW)

    Device RA Status                      : Ready           (RW)
    Device Link Status                    : Not Ready       (NR)
    Time of Last Device Link Status Change : N/A

    Device Suspend State                  : Offline
    Device Consistency State              : Disabled
    Device Consistency Exempt State       : Disabled
    RDF R2 Not Ready If Invalid           : Disabled
    Device Write Pacing Exempt State      : Disabled
    Effective Write Pacing Exempt State   : Disabled

    Device RDF State                      : Ready           (RW)
    Remote Device RDF State               : Write Disabled  (WD)

      RDF Pair State ( R1 <-  -> R2 )       : Suspended

    Number of R1 Invalid Tracks           : 0
    Number of R2 Invalid Tracks           : 0

    RDFA Information:
        {
        Session Number                    : 107
        Cycle Number                      : 0
        Number of Devices in the Session  : 8
        Session Status                    : Inactive
        Consistency Exempt Devices        : No
        Write Pacing Exempt Devices       : No

        Session Consistency State         : N/A
        Minimum Cycle Time                : 00:00:15
           Average Cycle Time               : 00:00:00
        Duration of Last cycle            : 00:00:00
        Session Priority                  : 33

        Tracks not Committed to the R2 Side: 0
```

```
  Time that R2 is behind R1          : 00:00:00
R2 Image Capture Time              : N/A
R2 Data is Consistent              : N/A
R1 Side Percent Cache In Use       : 0
R2 Side Percent Cache In Use       : 0

Transmit Idle Time                 : 00:00:00
R1 Side DSE Used Tracks            : 0
R2 Side DSE Used Tracks            : 0
R1 Side Shared Tracks              : 0
}
```

- The following query demonstrates the state of the SRDF devices and their SRDF links. Under normal circumstances, the SRDF pair is Synchronized (as shown below). The R1 devices are read-writeable and the SRDF links are read-writeable. However, the R2 devices, which are acting as mirrors to the R1 devices, are write disabled (WD) and cannot be written to by the target-side host at this time. The link is operating in synchronous replication (indicated by an S in the M column).

symrdf -g Rdf1Grp query

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW       0        0 RW 0109 WD       0        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf` command splits the SRDF pairs in the device group. As the `split` occurs, the singular SRDF control operations `suspend` and `rw_enable r2` occur. When the `split` is complete, a query reveals the altered state of the links and the R2 devices:

symrdf -g Rdf1Grp split -noprompt

```
An RDF 'Split' operation execution is
in progress for device group 'Rdf1Grp'. Please wait...

The RDF 'Split' operation successfully executed for
device group 'Rdf1Grp'.
```

- The following query shows the links are logically set to **NR** (not ready) and the R2 devices changed from WD to **RW**:

symrdf -g Rdf1Grp query

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 WD       0        0 NR 0108 RW       0        0 S...  Failed Over
DEV002  0109 WD       0        0 NR 0109 RW       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf establish` command performs an incremental establish for the SRDF pairs in the `Rdf1Grp` device group. It copies any changes made to the R1 devices while the devices were split to the R2 devices. Like all SRDF control operations, you can initiate the `establish` action from either the source or target side with the same results. The individual operations that combine to create an `establish` action are logged as they occur. For a more detailed report, examine the log file in `/var/symapi/log/symapi-yyyymmdd.log`.

symrdf -g Rdf1Grp establish -noprompt

```
n RDF 'Incremental Establish' operation execution is
in progress for device group 'Rdf1Grp'. Please wait...

    Write Disable device(s) on RA at target (R2)..............Done.
    Suspend RDF link(s)........................................Done.
    Resume RDF link(s)........................................Started.
    Merge device track tables between source and target.......Started.
    Devices: 0108-0109 in (0282,108)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)........................................Done.

The RDF 'Incremental Establish' operation successfully initiated for
device group 'Rdf1Grp'.
```

- An immediate `symrdf verify` command shows the SRDF pairs are not synchronized. The `echo $status` ($status for csh and $? for other shells) value of 5 is the code number that indicates no devices are synchronized. Because the devices are in the process of synchronizing, verifying the SyncInProg state returns a zero value that indicates success.

  symrdf -g Rdf1Grp verify -synchronized

```
None of the devices in the RDF group 'Rdf1Grp' are in the
'Synchronized' state.

echo $status
5
```

symrdf -g Rdf1Grp verify -syncinprog

```
All devices in the RDF group ' Rdf1Grp' are in the 'SyncInProg' state.

echo $status
0
```

- After some time elapses, the following `symrdf verify` command shows one of the SRDF pairs is fully synchronized and one is still in the process of synchronizing. As of this query snapshot, 108 remote (R2) invalid tracks on the source (R1) side still remain to be copied to the target device (108) to complete the synchronization process. The remote (R2) invalid tracks on the R1 side represent those tracks that are still owed to the R2 side.
- The `symrdf verify` command displays a message every 30 seconds until both SRDF pairs in the group are synchronized:

symrdf -g Rdf1Grp verify –i 30 -synchronized

```
Not all devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.

Not all devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.

All devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.
```

- Examine the return codes from the following `symrdf verify` commands. While `verify` and `verify –synchronized` return the success code 0, attempting to verify other states returns the appropriate failure code:

symrdf -g Rdf1Grp verify

```
All devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.

echo $status
0
```

symrdf -g Rdf1Grp -synchronized verify

```
All devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.

echo $status
0
```

symrdf -g Rdf1Grp -failedover verify

```
None of the devices in the RDF group 'Rdf1Grp' are in the 'Failed Over' state.

echo $status
34
```

symrdf -g Rdf1Grp -syncinprog verify

```
None of the devices in the RDF group 'Rdf1Grp' are in the 'SyncInProg' state.

echo $status
28
```

symrdf -g Rdf1Grp -split verify

```
None of the devices in the RDF group 'Rdf1Grp' are in the 'Split' state.

echo $status
26
```

- Both SRDF pairs in the device group are fully synchronized. The following `symrdf split` commands split the SRDF pairs in stages to show the values returned by the `symrdf verify -split` commands.

symrdf -g Rdf1Grp split DEV001 -noprompt

```
An RDF 'Split' operation execution is in progress for device 'DEV001' in
   device group 'Rdf1Grp'. Please wait...

The RDF 'Split' operation successfully executed for device 'DEV001' in
   device group 'Rdf1Grp'.
```

symrdf -g Rdf1Grp -split verify

```
Not all devices in the RDF group 'Rdf1Grp' are in the 'Split' state.

echo $status
25
```

- Once the previous split operation completes, the following `symrdf split` command can successfully split the other SRDF pair in the group. The subsequent `symrdf verify -split` command returns the success value of zero.

symrdf -g Rdf1Grp split -noprompt

```
An RDF 'Split' operation execution is in progress for device group 'Rdf1Grp'.
   Please wait...

The RDF 'Split' operation successfully executed for device group 'Rdf1Grp'.
```

symrdf -g Rdf1Grp -split verify

```
All devices in the RDF group 'Rdf1Grp' are in the 'Split' state.

echo $status
0
```

- When you initiate an SRDF control operation, the system checks the state of each SRDF pair involved in the operation. If a pair is not in an SRDF pair state that is valid (legal) for that operation, the operation fails unless the `-force` option is used with the command. The following command without `-force` rejects the failover operation because the SRDF pairs are currently in the Split state, which is not a legal state for a failover.

symrdf -g Rdf1Grp -noprompt failover

```
An RDF 'Failover' operation execution is in progress for device group 'Rdf1Grp'.
   Please wait...

Cannot proceed because the device pair is not in a legal RDF state.
```

- The following command with the `-force` option forces the failover operation to occur despite the unexpected Split state:

symrdf -g Rdf1Grp -noprompt –force failover

```
An RDF 'Failover' operation execution is in progress for device group 'Rdf1Grp'.
   Please wait...

    Write Disable device(s) on SA at source (R1)..............Done.
    Suspend RDF link(s).......................................Done.
    Read/Write Enable device(s) on RA at target (R2)..........Done.

The RDF 'Failover' operation successfully executed for device group 'Rdf1Grp'.
```

- The following query displays the results of the failover operation. The R1 devices in each SRDF pair are write disabled (WD), the SRDF links are suspended (NR), and the R2 devices are read/write (RW):

symrdf -g Rdf1Grp query

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 WD       0        0 NR 0108 RW       0        0 S...  Failed Over
DEV002  0109 WD       0        0 NR 0109 RW       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The `symrdf failback` command initiates a failback on one of the two SRDF pairs in the `Rdf1Grp` device group:

symrdf -g Rdf1Grp failback DEV001 -noprompt

```
An RDF 'Failback' operation execution is in progress for device
'DEV001' in group 'Rdf1Grp'. Please wait...

    Write Disable device(s) on RA at target (R2)..............Done.
    Suspend RDF link(s).......................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0108-0108 in (0282,108)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)........................................Started.
    Resume RDF link(s)........................................Done.
    Read/Write Enable device(s) on SA at source (R1)..........Done.

The RDF 'Failback' operation successfully executed for device
'DEV001' in group 'Rdf1Grp'.
```

- The following query displays the states of the SRDF pairs in the device group. The two SRDF pairs are now in different states: one pair is in the Synchronized state and the other pair is still in the Failed Over state:

```
symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 WD       0        0 NR 0109 RW       0        0 S...  Failed Over

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- This subsequent `symrdf failback` command initiates a failback on the SRDF pair that is still in the Failed Over state:

```
symrdf -g Rdf1Grp failback DEV002 -noprompt

An RDF 'Failback' operation execution is in progress for device
'DEV002' in group 'Rdf1Grp'. Please wait...

    Write Disable device(s) on RA at target (R2)..............Done.
    Suspend RDF link(s)......................................Done.
    Merge device track tables between source and target.......Started.
    Devices: 0109-0109 in (0282,108)......................... Merged.
    Merge device track tables between source and target.......Done.
    Resume RDF link(s)......................................Started.
    Resume RDF link(s)......................................Done.
    Read/Write Enable device(s) on SA at source (R1)..........Done.

The RDF 'Failback' operation successfully executed for device
```

- The following query once again displays the states of the SRDF pairs in the `Rdf1Grp` device group. The two SRDF pairs are now in complementary states — one pair is in the Synchronized state and the other pair is in the SyncInProg state:

```
symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)
```

```
        Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
              ST                  LI      ST
Standard       A                   N       A
Logical        T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev    E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW    1048        0 RW 0109 WD    1048        0 S...  SyncInProg

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- Because there are different SRDF pair states in the device group, the following `symdg show` command displays `Mixed` as the state of the SRDF pairs in the group:

symdg show Rdf1Grp

```
Group Name:  Rdf1Grp

    Group Type                                  : RDF1     (RDFA)
    Device Group in GNS                         : No
    Valid                                       : Yes
    Symmetrix ID                                : 000192600282
    Group Creation Time                         : Tue Apr 12 16:04:16 2011
    Vendor ID                                   : EMC Corp
    Application ID                              : SYMCLI

    Number of STD Devices in Group              :    2
    Number of Associated GK's                   :    0
    Number of Locally-associated BCV's          :    0
    Number of Locally-associated VDEV's         :    0
    Number of Locally-associated TGT's          :    0
    Number of Remotely-associated VDEV's(STD RDF):    0
    Number of Remotely-associated BCV's (STD RDF):    0
    Number of Remotely-associated TGT's(TGT RDF) :    0
    Number of Remotely-associated BCV's (BCV RDF):    0
    Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
    Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
    Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
    Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0
    Number of Composite Groups                  :    0
    Composite Group Names                       : N/A

    Standard (STD) Devices (2):
        {
        ------------------------------------------------------------------------------------------------
                                                                                Sym  Device                    Cap
        LdevName              PdevName                                         Dev  Config        Att. Sts     (MB)
                                ----------------------------------------------------------------------------
        DEV001                /dev/rdsk/c56t10d4                               0108 RDF1+Mir           RW      4314
        DEV002                /dev/rdsk/c56t10d5                               0109 RDF1+Mir           RW      4314
        }

    Device Group RDF Information
        {
        RDF Type                            : R1
        RDF (RA) Group Number               : 108              (6B)

        Remote Symmetrix ID                 : 000192603139

        R2 Device Is Larger Than The R1 Device : False

        Paired with a Diskless Device       : False
        Paired with a Concurrent Device     : False
        Paired with a Cascaded Device       : False
        Thick Thin Relationship             : False

         RDF Pair Configuration              : Normal
```

```
RDF STAR Mode                          : False

RDF Mode                               : Synchronous
RDF Adaptive Copy                      : Disabled
RDF Adaptive Copy Write Pending State  : N/A
RDF Adaptive Copy Skew (Tracks)        : 32767

RDF Device Domino                      : Disabled

RDF Link Configuration                 : Fibre
RDF Link Domino                        : Disabled
Prevent Automatic RDF Link Recovery    : Enabled
Prevent RAs Online Upon Power ON       : Enabled

Device RDF Status                      : Ready           (RW)

Device RA Status                       : Ready           (RW)
Device Link Status                     : Ready           (RW)
Time of Last Device Link Status Change : N/A

Device Suspend State                   : N/A
Device Consistency State               : Disabled
Device Consistency Exempt State        : Disabled
RDF R2 Not Ready If Invalid            : Disabled
Device Write Pacing Exempt State       : Disabled
Effective Write Pacing Exempt State    : Disabled

Device RDF State                       : Ready           (RW)
 Remote Device RDF State                : Write Disabled  (WD)

RDF Pair State (  R1 <===> R2 )        : Mixed

Number of R1 Invalid Tracks            : 0
Number of R2 Invalid Tracks            : 1048

RDFA Information:
       {
    Session Number                     : 107
       Cycle Number                       : 0
    Number of Devices in the Session   : 8
    Session Status                     : Inactive
    Consistency Exempt Devices         : No
    Write Pacing Exempt Devices        : No

    Session Consistency State          : N/A
    Minimum Cycle Time                 : 00:00:15
    Average Cycle Time                 : 00:00:00
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33

    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:00
       R2 Image Capture Time              : N/A
    R2 Data is Consistent              : N/A
    R1 Side Percent Cache In Use       : 0
    R2 Side Percent Cache In Use       : 0

    Transmit Idle Time                 : 00:00:00
    R1 Side DSE Used Tracks            : 0
    R2 Side DSE Used Tracks            : 0
    R1 Side Shared Tracks              : 0
```

- As a prerequisite for associating RDF1 BCV devices with a device group, the `symdev list` command with the `-r1 -bcv` options displays all RDF1 BCV devices. Devices not associated with a device group (N/Asst'd) are free to be added to the `Rdf1Grp` device group.

`symdev list -sid 282 -r1 -bcv`

```
Symmetrix ID: 000192600282

        Device Name                                   Directors                  Device
------------------------------------------------- ------------- ------------------------------------
                                                                                                Cap
Sym  Physical                                     SA :P DA :IT  Config        Attribute    Sts   (MB)
------------------------------------------------- ------------- ------------------------------------

010E /dev/rdsk/c56t11d2                           08E:1 07A:C4  RDF1-BCV+Mir  N/Asst'd     RW    4314
010F /dev/rdsk/c56t11d3                           08E:1 09C:D3  RDF1-BCV+Mir  N/Asst'd     RW    4314
```

- The following `symbcv` commands associate two of these RDF1 BCV devices (10e and 10f) with the `Rdf1Grp` device group. SYMCLI assigns the devices the default logical names BCV001 and BCV002, respectively.

```
symbcv -g Rdf1Grp -sid 282 associate dev 10e
symbcv -g Rdf1Grp -sid 282 associate dev 10f
```

- The following `symrdf query` command without options displays only the SRDF pair state of the SRDF standard devices in the `Rdf1Grp` device group:

```
symrdf -g Rdf1Grp query

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW    1048        0 RW 0109 WD    1048        0 S...  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The `symrdf query` command with the `-bcv` option displays only the SRDF pair state of the SRDF BCV devices in the device group:

```
symrdf -g Rdf1Grp query -bcv

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
BCV           A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

BCV001  010E RW       0        0 NR 010E WD       0        0 C.D.  Suspended
BCV002  010F RW       0        0 NR 010F WD       0        0 C.D.  Suspended

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:
```

```
 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The `symrdf query` command with the `-all` option displays the SRDF pair state of all devices in the device group, regardless of the device type:

symrdf -g Rdf1Grp query -all

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW       0        0 RW 0109 WD       0        0 S...  Synchronized
BCV001  010E RW       0        0 NR 010E WD       0        0 C.D.  Suspended
BCV002  010F RW       0        0 NR 010F WD       0        0 C.D.  Suspended

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The `symmir establish` command creates BCV pairs. The `-exact` option matches standard devices with BCV devices in the exact order in which they were added to the device group. In this example, device 108 is established with device 10e, and device 109 is established with device 10f:

symmir -g Rdf1Grp establish -full -exact -noprompt

```
'Full Establish' operation execution is in progress for device group 'Rdf1Grp'.
   Please wait...

'Full Establish' operation successfully initiated for device group 'Rdf1Grp'.
```

- The `symmir query` command displays the BCV pairs in the device group and their state of mirroring:

symmir -g Rdf1Grp query

```
Device Group (DG) Name: Rdf1Grp
DG's Type             : RDF1
DG's Symmetrix ID     : 000192600282


     Standard Device                    BCV Device                  State
------------------------- ------------------------------------- ------------
```

```
                    Inv.                                  Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- ------------

DEV001         0108      0 BCV001               010E *         0 Synchronized
DEV002         0109      0 BCV002               010F *         0 Synchronized

Total              -------                                -------
  Track(s)               0                                      0
  MB(s)                0.0                                    0.0

Legend:

(*): The paired BCV device is associated with this group.
```

- The `symrdf query -all` command displays the SRDF pair state of all SRDF devices in the device group. While established with DEV001 and DEV002 (in other words, as part of a BCV pair), the RDF1 BCV devices are in a Suspended pair state and cannot copy data to their respective target devices (10e and 10f):

symrdf -g Rdf1Grp query -all

```
Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW       0        0 RW 0109 WD       0        0 S...  Synchronized
BCV001  010E NR       0        0 NR 010E WD       0        0 C.D.  Suspended
BCV002  010F NR       0        0 NR 010F WD       0        0 C.D.  Suspended

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symdg show` command displays group information about the `Rdf1Grp` device group, which contains two RDF1 standard devices and two RDF1 BCV devices. The Device Group RDF Information section of the display shows that the composite SRDF pair state of the RDF1 standard devices is Synchronized. The Device Group BCV RDF Information section of the display shows that the composite SRDF pair state of the RDF1 BCV devices is Suspended.

symdg show Rdf1Grp

```
Group Name:  Rdf1Grp

    Group Type                                  : RDF1     (RDFA)
    Device Group in GNS                         : No
    Valid                                       : Yes
    Symmetrix ID                                : 000192600282
    Group Creation Time                         : Tue Apr 12 16:04:16 2011
    Vendor ID                                   : EMC Corp
    Application ID                              : SYMCLI

    Number of STD Devices in Group              :    2
    Number of Associated GK's                   :    0
    Number of Locally-associated BCV's          :    2
    Number of Locally-associated VDEV's         :    0
    Number of Locally-associated TGT's          :    0
    Number of Remotely-associated VDEV's(STD RDF):    0
    Number of Remotely-associated BCV's (STD RDF):    0
    Number of Remotely-associated TGT's(TGT RDF) :    0
    Number of Remotely-associated BCV's (BCV RDF):    0
    Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0
    Number of Remotely-assoc'd BCV's (Hop-2 BCV) :    0
    Number of Remotely-assoc'd VDEV's(Hop-2 VDEV):    0
    Number of Remotely-assoc'd TGT's (Hop-2 TGT) :    0
    Number of Composite Groups                  :    0
    Composite Group Names                       : N/A

    Standard (STD) Devices (2):
        {
        ----------------------------------------------------------------------------------------------------
                                                                      Sym  Device                      Cap
        LdevName              PdevName                                Dev  Config        Att. Sts      (MB)
                                ----------------------------------------------------------------------------
        DEV001                /dev/rdsk/c56t10d4                      0108 RDF1+Mir           RW      4314
        DEV002                /dev/rdsk/c56t10d5                      0109 RDF1+Mir           RW      4314
        }

    BCV Devices Locally-associated (2):
        {
        ----------------------------------------------------------------------------------------------------
                                                                      Sym  Device                      Cap
        LdevName              PdevName                                Dev  Config        Att. Sts      (MB)
        ----------------------------------------------------------------------------------------------------
        BCV001                /dev/rdsk/c56t11d2                      010E RDF1-BCV+Mir       NR      4314
        BCV002                /dev/rdsk/c56t11d3                      010F RDF1-BCV+Mir       NR      4314
        }

    Device Group RDF Information
        {
        RDF Type                              : R1
        RDF (RA) Group Number                 : 108               (6B)

        Remote Symmetrix ID                   : 000192603139

        R2 Device Is Larger Than The R1 Device : False

        Paired with a Diskless Device         : False
        Paired with a Concurrent Device       : False
        Paired with a Cascaded Device         : False
        Thick Thin Relationship               : False

        RDF Pair Configuration                : Normal
        RDF STAR Mode                         : False

        RDF Mode                              : Synchronous
        RDF Adaptive Copy                     : Disabled
        RDF Adaptive Copy Write Pending State : N/A
        RDF Adaptive Copy Skew (Tracks)       : 32767

        RDF Device Domino                     : Disabled

        RDF Link Configuration                : Fibre
        RDF Link Domino                       : Disabled
        Prevent Automatic RDF Link Recovery   : Enabled
        Prevent RAs Online Upon Power ON      : Enabled

        Device RDF Status                     : Ready           (RW)

        Device RA Status                      : Ready           (RW)
        Device Link Status                    : Ready           (RW)
        Time of Last Device Link Status Change : N/A

        Device Suspend State                  : N/A
        Device Consistency State              : Disabled
        Device Consistency Exempt State       : Disabled
          RDF R2 Not Ready If Invalid           : Disabled
        Device Write Pacing Exempt State      : Disabled
        Effective Write Pacing Exempt State   : Disabled

        Device RDF State                      : Ready           (RW)
          Remote Device RDF State               : Write Disabled  (WD)
```

```
    RDF Pair State (  R1 <===> R2 )        : Synchronized

    Number of R1 Invalid Tracks            : 0
    Number of R2 Invalid Tracks            : 0

    RDFA Information:
        {
        Session Number                     : 107
        Cycle Number                       : 0
        Number of Devices in the Session   : 8
        Session Status                     : Inactive
        Consistency Exempt Devices         : No
        Write Pacing Exempt Devices        : No

        Session Consistency State          : N/A
           Minimum Cycle Time                : 00:00:15
        Average Cycle Time                 : 00:00:00
        Duration of Last cycle             : 00:00:00
        Session Priority                   : 33

        Tracks not Committed to the R2 Side: 0
        Time that R2 is behind R1          : 00:00:00
        R2 Image Capture Time              : N/A
        R2 Data is Consistent              : N/A
        R1 Side Percent Cache In Use       : 0
        R2 Side Percent Cache In Use       : 0

        Transmit Idle Time                 : 00:00:00
        R1 Side DSE Used Tracks            : 0
        R2 Side DSE Used Tracks            : 0
        R1 Side Shared Tracks              : 0

Device Group BCV RDF Information
    {
    RDF Type                               : R1
    RDF (RA) Group Number                  : 108               (6B)

    Remote Symmetrix ID                    : 000192603139

    R2 Device Is Larger Than The R1 Device : False

    Paired with a Diskless Device          : False
    Paired with a Concurrent Device        : False
    Paired with a Cascaded Device          : False
    Thick Thin Relationship                : False

    RDF Pair Configuration                 : Normal
    RDF STAR Mode                          : False

    RDF Mode                               : Adaptive Copy
    RDF Adaptive Copy                      : Enabled: Disk Mode
    RDF Adaptive Copy Write Pending State  : N/A
    RDF Adaptive Copy Skew (Tracks)        : 32767

    RDF Device Domino                      : Disabled

    RDF Link Configuration                 : Fibre
    RDF Link Domino                        : Disabled
    Prevent Automatic RDF Link Recovery    : Enabled
    Prevent RAs Online Upon Power ON       : Enabled

    Device RDF Status                      : Ready           (RW)

    Device RA Status                       : Ready           (RW)
    Device Link Status                     : Not Ready       (NR)
    Time of Last Device Link Status Change : N/A

    Device Suspend State                   : Offline
    Device Consistency State               : Disabled
    Device Consistency Exempt State        : N/A
    RDF R2 Not Ready If Invalid            : Mixed
    Device Write Pacing Exempt State       : Enabled
    Effective Write Pacing Exempt State    : Enabled

    Device RDF State                       : Not Ready       (NR)
    Remote Device RDF State                : Write Disabled  (WD)

    RDF Pair State ( R1 <-  -> R2 )        : Suspended

    Number of R1 Invalid Tracks            : 0
    Number of R2 Invalid Tracks            : 0

    RDFA Information:
        {
        Session Number                     : 107
        Cycle Number                       : 0
        Number of Devices in the Session   : 8
        Session Status                     : Inactive
        Consistency Exempt Devices         : No
        Write Pacing Exempt Devices        : No

        Session Consistency State          : N/A
        Minimum Cycle Time                 : 00:00:15
        Average Cycle Time                 : 00:00:00
```

```
Duration of Last cycle            : 00:00:00
Session Priority                  : 33

Tracks not Committed to the R2 Side: 0
Time that R2 is behind R1         : 00:00:00
R2 Image Capture Time             : N/A
R2 Data is Consistent             : N/A
R1 Side Percent Cache In Use      : 0
R2 Side Percent Cache In Use      : 0

Transmit Idle Time                : 00:00:00
   R1 Side DSE Used Tracks           : 0
R2 Side DSE Used Tracks           : 0
R1 Side Shared Tracks             : 0
```

- The `symmir split` command splits the BCV pairs in the Rdf1Grp device group. When the split completes, the SRDF links for the RDF1 BCV devices are still not ready (NR), even though the state of the source (R1) RDF1 BCV devices changed from not ready to read/write enabled.

```
symmir -g Rdf1Grp split -noprompt

'Split' operation execution is in progress for device group 'Rdf1Grp'.
   Please wait...

'Split' operation successfully executed for device group 'Rdf1Grp'.
```

- The following `symrdf query –bcv` command shows the SRDF pair state of the RDF1 BCV devices in the device group. Because the links for those devices are not ready (read/write disabled), the SRDF pair state remains Suspended. On the source (R1) side, each RDF1 BCV device has 276 remote (R2) invalid tracks to be copied to the BCV's remote (R2) mirror when synchronization begins.

```
symrdf -g Rdf1Grp query -bcv

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)


       Source (R1) View                 Target (R2) View     MODES
--------------------------------    ------------------------ ----- ------------
             ST                  LI      ST
BCV           A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

BCV001  010E NR       0      276 NR 010E WD       0        0 C.D.  Suspended
BCV002  010F NR       0      276 NR 010F WD       0        0 C.D.  Suspended

Total          -------- --------           -------- --------
  Track(s)            0      552                  0        0
  MB(s)             0.0     34.5                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf establish` command with the `-bcv` option resumes the SRDF links for the RDF1 BCV devices and initiates the propagation of data from the source (R1) RDF1 BCV devices to their remote (R2) mirror devices:

```
symrdf -g Rdf1Grp establish -bcv -noprompt

An RDF 'Incremental Establish' operation execution is in progress for device
   group 'Rdf1Grp'. Please wait...

    Suspend RDF link(s).......................................Done.
    Resume RDF link(s).......................................Done.

The RDF 'Incremental Establish' operation successfully initiated for device
   group 'Rdf1Grp'.
```

- The following `symrdf query -bcv` command once again shows the SRDF pair state of the RDF1 BCV devices (now SyncInProg) and the number of remote (R2) invalid tracks on the source (R1) side that still need to be copied to the BCVs' remote (R2) mirrors to complete the synchronization process. Note the establish operation changed the state of the links from NR (not ready) to RW (read/write).

```
symrdf -g Rdf1Grp query -bcv

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)
Source (R1) View                 Target (R2) View     MODES
--------------------------------   ------------------------ ----- ------------
             ST                  LI      ST
BCV           A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

BCV001  010E RW       0      276 RW 010E WD       0        0 C.D.  SyncInProg
BCV002  010F RW       0      276 RW 010F WD       0        0 C.D.  SyncInProg

Total          -------- --------           -------- --------
  Track(s)            0      552                  0        0
  MB(s)             0.0     34.5                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following `symrdf verify -all` command checks the state of all SRDF pairs in the `Rdf1Grp` device group every five seconds until all SRDF pairs are synchronized:

```
symrdf -g Rdf1Grp verify -all -i 5 -synchronized

NOT all of the mirrored pairs are in the 'Synchronized' state.

NOT all of the mirrored pairs are in the 'Synchronized' state.

All devices in the RDF group 'Rdf1Grp' are in the 'Synchronized' state.
```

- The `symrdf query -all` command displays all SRDF devices and their states. Like the RDF1 standard devices, the RDF1 BCV devices are now in the Synchronized SRDF pair state. The copying of data from the source (R1) side to the target (R2) side is complete.

```
symrdf -g Rdf1Grp query -all

Device Group (DG) Name             : Rdf1Grp
DG's Type                          : RDF1
DG's Symmetrix ID                  : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID                : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number              : 108 (6B)

Source (R1) View                 Target (R2) View     MODES
--------------------------------   ------------------------ ----- ------------
             ST                  LI      ST
Standard      A                   N       A
Logical       T  R1 Inv   R2 Inv  K       T  R1 Inv   R2 Inv       RDF Pair
Device  Dev   E  Tracks   Tracks  S Dev   E  Tracks   Tracks MDAE  STATE
-------------------------------- -- ------------------------ ----- ------------

DEV001  0108 RW       0        0 RW 0108 WD       0        0 S...  Synchronized
DEV002  0109 RW       0        0 RW 0109 WD       0        0 S...  Synchronized
BCV001  010E RW       0        0 RW 010E WD       0        0 C.D.  Synchronized
BCV002  010F RW       0        0 RW 010F WD       0        0 C.D.  Synchronized

Total          -------- --------           -------- --------
  Track(s)            0        0                  0        0
  MB(s)             0.0      0.0                0.0      0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

Example 2: Querying a composite group

Querying a composite group is similar to querying a device group, except that the symrdf query command includes the –cg option and the name of the composite group.

The hardware setup consists of a Solaris host connected to two source Symmetrix arrays (Symmetrix 282 and Symmetrix 088). The example builds a composite group with source R1 devices from both Symmetrix arrays and enables consistency protection for the composite group. For more examples using SRDF consistency protection, refer to "Implementing Consistency Protection" on page 377.

- The following `symcg create` command creates an RDF1 type composite group named `SRDF` enabled for consistency protection:

  ```
  symcg create SRDF -type rdf1 -rdf_consistency
  ```

- The following `symcg addall` command adds a range of standard devices from Symmetrix 282 to the `SRDF` composite group:

  ```
  symcg -cg SRDF addall dev -devs 100:107 -sid 282
  ```

- The following `symcg addall` command adds a range of standard devices from Symmetrix 088 to the `SRDF` composite group:

  ```
  symcg -cg SRDF addall dev -devs 100:107 -sid 088
  ```

- The following query checks the state of the SRDF pairs. The SRDF pairs from one Symmetrix array are in the Synchronized state while the other Symmetrix array has SRDF pairs in the Suspended state.

symrdf -cg SRDF query

```
Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  2
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number             : 104 (67)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   0100 RW       0       0 RW 0100 WD       0       0 S...  .  -   Synchronized
DEV002   0101 RW       0       0 RW 0101 WD       0       0 S...  .  -   Synchronized
DEV003   0102 RW       0       0 RW 0102 WD       0       0 S...  .  -   Synchronized
DEV004   0103 RW       0       0 RW 0103 WD       0       0 S...  .  -   Synchronized
DEV005   0104 RW       0       0 RW 0104 WD       0       0 S...  .  -   Synchronized
DEV006   0105 RW       0       0 RW 0105 WD       0       0 S...  .  -   Synchronized
DEV007   0106 RW       0       0 RW 0106 WD       0       0 S...  .  -   Synchronized
DEV008   0107 RW       0       0 RW 0107 WD       0       0 S...  .  -   Synchronized

Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number             : 105 (68)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV009   0100 RW       0       0 NR 0100 WD       0       0 S...  .  .   Suspended
DEV010   0101 RW       0       0 NR 0101 WD       0       0 S...  .  .   Suspended
DEV011   0102 RW       0       0 NR 0102 WD       0       0 S...  .  .   Suspended
DEV012   0103 RW       0       0 NR 0103 WD       0       0 S...  .  .   Suspended
DEV013   0104 RW       0       0 NR 0104 WD       0       0 S...  .  .   Suspended
DEV014   0105 RW       0       0 NR 0105 WD       0       0 S...  .  .   Suspended
DEV015   0106 RW       0       0 NR 0106 WD       0       0 S...  .  .   Suspended
DEV016   0107 RW       0       0 NR 0107 WD       0       0 S...  .  .   Suspended

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:
```

```
 Cons(istency State)    : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)        : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The `symrdf establish` command initiates an incremental establish operation on the SRDF pairs in the `SRDF` composite group that are not synchronized (that is, the suspended pairs on Symmetrix 0282):

```
symrdf -cg SRDF establish -noprompt

An RDF 'Incremental Establish' operation execution is
in progress for composite group 'SRDF'. Please wait...

    Suspend RDF link(s) for device(s) in (0282,104)..................Done.
    Resume RDF link(s) for device(s) in (0282,104)...................Started.
    Resume RDF link(s) for device(s) in (0282,104)...................Done.

The RDF 'Incremental Establish' operation successfully initiated for
composite group 'SRDF'.
```

- Another query shows the previously suspended pairs are now in the process of synchronizing. A period (.) in the **Cons** column indicates that consistency protection is disabled.

```
symrdf -cg SRDF query

Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  2
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                     : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID              : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number            : 104 (67)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv      n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   0100 RW       0   65036 RW 0100 WD       0       0 S...  .  -   SyncInProg
DEV002   0101 RW       0   65053 RW 0101 WD       0       0 S...  .  -   SyncInProg
DEV003   0102 RW       0   64217 RW 0102 WD       0       0 S...  .  -   SyncInProg
DEV004   0103 RW       0   66511 RW 0103 WD       0       0 S...  .  -   SyncInProg
DEV005   0104 RW       0   64857 RW 0104 WD       0       0 S...  .  -   SyncInProg
DEV006   0105 RW       0   66464 RW 0105 WD       0       0 S...  .  -   SyncInProg
DEV007   0106 RW       0   63766 RW 0106 WD       0       0 S...  .  -   SyncInProg
DEV008   0107 RW       0   63922 RW 0107 WD       0       0 S...  .  -   SyncInProg

Symmetrix ID                     : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID              : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number            : 105 (68)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv      n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV009   0100 RW       0       0 RW 0100 WD       0       0 S...  .  -   Synchronized
DEV010   0101 RW       0       0 RW 0101 WD       0       0 S...  .  -   Synchronized
```

```
DEV011   0102 RW       0       0 RW 0102 WD       0       0 S...  .  -   Synchronized
DEV012   0103 RW       0       0 RW 0103 WD       0       0 S...  .  -   Synchronized
DEV013   0104 RW       0       0 RW 0104 WD       0       0 S...  .  -   Synchronized
DEV014   0105 RW       0       0 RW 0105 WD       0       0 S...  .  -   Synchronized
DEV015   0106 RW       0       0 RW 0106 WD       0       0 S...  .  -   Synchronized
DEV016   0107 RW       0       0 RW 0107 WD       0       0 S...  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0  519826                  0       0
  MBs                0.0 32489.1                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - =
```

- The following `symcg show` command confirms that consistency protection for device pairs in the `SRDF` composite group is currently disabled (indicated by a period under the Flags C column):

symcg show SRDF

```
Composite Group Name:  SRDF

  Composite Group Type                              : RDF1
  Valid                                             : Yes
  CG in PowerPath                                   : No
  CG in GNS                                         : Yes
  RDF Consistency Protection Allowed                : Yes
  RDF Consistency Mode                              : NONE
  Concurrent RDF                                    : No
  Cascaded RDF                                      : No

  Number of RDF (RA) Groups                         :    2
  Number of STD Devices                             :   16
  Number of CRDF STD Devices                        :    0
  Number of BCV's (Locally-associated)              :    0
  Number of VDEV's (Locally-associated)             :    0
  Number of TGT's Locally-associated                :    0
  Number of CRDF TGT Devices                        :    0
  Number of RVDEV's (Remotely-associated VDEV)      :    0
  Number of RBCV's (Remotely-associated STD-RDF)    :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)   :    0
  Number of RRBCV's (Remotely-associated RBCV)      :    0
  Number of RTGT's (Remotely-associated)            :    0
  Number of Hop2 BCV's (Remotely-assoc'ed Hop2 BCV) :    0
  Number of Hop2 VDEV's (Remotely-assoc'ed Hop2 VDEV):   0
  Number of Hop2 TGT's (Remotely-assoc'ed Hop2 TGT) :    0
  Number of Device Groups                           :    0
  Device Group Names                                : N/A

  Number of Symmetrix Units (2):
    {

    1) Symmetrix ID                                 : 000192600282
       Microcode Version                            : 5875
       Number of STD Devices                        :    8
       Number of CRDF STD Devices                   :    0
        Number of BCV's (Locally-associated)         :    0
       Number of VDEV's (Locally-associated)        :    0
       Number of TGT's Locally-associated           :    0
       Number of CRDF TGT Devices                   :    0
       Number of RVDEV's (Remotely-associated VDEV) :    0
       Number of RBCV's (Remotely-associated STD_RDF) :  0
       Number of BRBCV's (Remotely-associated BCV-RDF):  0
       Number of RTGT's (Remotely-associated)       :    0
       Number of RRBCV's (Remotely-associated RBCV) :    0
       Number of Hop2BCV's (Remotely-assoc'ed Hop2BCV):  0
       Number of Hop2VDEVs(Remotely-assoc'ed Hop2VDEV):  0
       Number of Hop2TGT's (Remotely-assoc'ed Hop2TGT):  0

       Number of RDF (RA) Groups (1):
            {
```

```
      1) RDF (RA) Group Number  : 104            (67)
         Remote Symmetrix ID    : 000192603139
         Microcode Version      : 5875
         Recovery RA Group      : N/A            (N/A)
           RA Group Name          : N/A

         STD Devices (8):
           {
           ------------------------------------------------------------------------------------------
                                                                Sym  Device             Flags     Cap
           LdevName    PdevName                                 Dev  Config        Sts   CSRT     (MB)
           ------------------------------------------------------------------------------------------
           DEV001      N/A                                      0100 RDF1+Mir      RW    .--1     4314
           DEV002      N/A                                      0101 RDF1+Mir      RW    .--1     4314
           DEV003      N/A                                      0102 RDF1+Mir      RW    .--1     4314
           DEV004      N/A                                      0103 RDF1+Mir      RW    .--1     4314
           DEV005      N/A                                      0104 RDF1+Mir      RW    .--1     4314
           DEV006      N/A                                      0105 RDF1+Mir      RW    .--1     4314
           DEV007      N/A                                      0106 RDF1+Mir      RW    .--1     4314
           DEV008      N/A                                      0107 RDF1+Mir      RW    .--1     4314
           }
      }

   2) Symmetrix ID                                  : 000192603088
      Microcode Version                             : 5875
      Number of STD Devices                         :    8
      Number of CRDF STD Devices                    :    0
      Number of BCV's (Locally-associated)          :    0
      Number of VDEV's (Locally-associated)         :    0
      Number of TGT's Locally-associated            :    0
      Number of CRDF TGT Devices                    :    0
      Number of RVDEV's (Remotely-associated VDEV)  :    0
      Number of RBCV's (Remotely-associated STD_RDF) :    0
      Number of BRBCV's (Remotely-associated BCV-RDF):    0
      Number of RTGT's (Remotely-associated)        :    0
      Number of RRBCV's (Remotely-associated RBCV)  :    0
      Number of Hop2BCV's (Remotely-assoc'ed Hop2BCV):    0
      Number of Hop2VDEVs(Remotely-assoc'ed Hop2VDEV):    0
      Number of Hop2TGT's (Remotely-assoc'ed Hop2TGT):    0

      Number of RDF (RA) Groups (1):
        {

      1) RDF (RA) Group Number  : 105            (68)
         Remote Symmetrix ID    : 000192603156
         Microcode Version      : 5875
         Recovery RA Group      : N/A            (N/A)
         RA Group Name          : N/A

         STD Devices (8):
           {
           ------------------------------------------------------------------------------------------
                                                                Sym  Device             Flags     Cap
           LdevName    PdevName                                 Dev  Config        Sts   CSRT     (MB)
           ------------------------------------------------------------------------------------------
           DEV009      N/A                                      0100 RDF1+Mir      RW    .--1     4314
           DEV010      N/A                                      0101 RDF1+Mir      RW    .--1     4314
           DEV011      N/A                                      0102 RDF1+Mir      RW    .--1     4314
           DEV012      N/A                                      0103 RDF1+Mir      RW    .--1     4314
           DEV013      N/A                                      0104 RDF1+Mir      RW    .--1     4314
           DEV014      N/A                                      0105 RDF1+Mir      RW    .--1     4314
           DEV015      N/A                                      0106 RDF1+Mir      RW    .--1     4314
           DEV016      N/A                                      0107 RDF1+Mir      RW    .--1     4314
           }
      }
    }

Legend:
  RDFA Flags:
       C(onsistency)   : X = Enabled, . = Disabled, - = N/A
       (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
       R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
       (Mirror) T(ype) : 1 = R1, 2 = R2, - = N/A
```

- The following `symcg enable` command enables consistency protection for device pairs in the `SRDF` composite group:

```
symcg -cg SRDF enable -noprompt

A consistency 'Enable' operation execution is in progress for composite group
   'SRDF'. Please wait...

The consistency 'Enable' operation successfully executed for composite group
   'SRDF'.
```

- Another query shows that all pairs are in the Synchronized state. As indicated in the Legend, an X in the Cons column indicates that all pairs are now enabled for consistency protection.

symrdf -cg SRDF query

```
Composite Group Name      :  SRDF
Composite Group Type      :  RDF1
Number of Symmetrix Units :  2
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number             : 105 (68)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV009   0100 RW       0       0 RW 0100 WD       0       0 S...  X  -   Synchronized
DEV010   0101 RW       0       0 RW 0101 WD       0       0 S...  X  -   Synchronized
DEV011   0102 RW       0       0 RW 0102 WD       0       0 S...  X  -   Synchronized
DEV012   0103 RW       0       0 RW 0103 WD       0       0 S...  X  -   Synchronized
DEV013   0104 RW       0       0 RW 0104 WD       0       0 S...  X  -   Synchronized
DEV014   0105 RW       0       0 RW 0105 WD       0       0 S...  X  -   Synchronized
DEV015   0106 RW       0       0 RW 0106 WD       0       0 S...  X  -   Synchronized
DEV016   0107 RW       0       0 RW 0107 WD       0       0 S...  X  -   Synchronized

Symmetrix ID                      : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number             : 104 (67)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   0100 RW       0       0 RW 0100 WD       0       0 S...  X  -   Synchronized
DEV002   0101 RW       0       0 RW 0101 WD       0       0 S...  X  -   Synchronized
DEV003   0102 RW       0       0 RW 0102 WD       0       0 S...  X  -   Synchronized
DEV004   0103 RW       0       0 RW 0103 WD       0       0 S...  X  -   Synchronized
DEV005   0104 RW       0       0 RW 0104 WD       0       0 S...  X  -   Synchronized
DEV006   0105 RW       0       0 RW 0105 WD       0       0 S...  X  -   Synchronized
DEV007   0106 RW       0       0 RW 0106 WD       0       0 S...  X  -   Synchronized
DEV008   0107 RW       0       0 RW 0107 WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:
```

```
 Cons(istency State)    : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)        : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Verifying if invalid tracks are owed to R1 from R2

This example shows how to use the `symrdf verify -consistent -noinvalids` command to check if the R2 side owes invalid tracks to the R1 side.

- The following two query commands, executed at the same time, show the device pairs in `HOP1` are in the Consistent state. However, the output shows local invalid tracks on the R1 side and remote invalid tracks on the R2 side.

symrdf -cg HOP1 query

```
Composite Group Name      :  HOP1
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE

Symmetrix ID                      : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number             :  98 (61)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------ ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ------------------------ ----- ------ ------------
DEV001   0060 RW   29631       0 RW 0060 WD   29623       0 A...  .  -   Consistent
DEV002   0061 RW   25215       0 RW 0061 WD   25145       0 A...  .  -   Consistent
DEV003   0062 RW   29299       0 RW 0062 WD   29294       0 A...  .  -   Consistent
DEV004   0063 RW   25549       0 RW 0063 WD   25463       0 A...  .  -   Consistent
DEV005   0064 RW   24360       0 RW 0064 WD   24285       0 A...  .  -   Consistent
DEV006   0065 RW   25527       0 RW 0065 WD   25459       0 A...  .  -   Consistent
DEV007   0066 RW   25193       0 RW 0066 WD   25121       0 A...  .  -   Consistent
DEV008   0067 RW   29880       0 RW 0067 WD   29874       0 A...  .  -   Consistent

Total            ------- -------            ------- -------
  Track(s)        214654       0             214264       0
  MBs            13415.9     0.0            13391.5     0.0
```

symrdf -cg HOP1 verify -consistent

```
All devices in the group 'HOP1' are in 'Consistent' state.
```

- The following query shows the `HOP1` device pairs are R2 Consistent even through there are invalid tracks:

symrdf -cg HOP1 query

```
Composite Group Name      :  HOP1
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE

Symmetrix ID                      : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number             :  98 (61)
```

```
        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
               ST                 LI      ST                       C  S
  Standard      A                  N       A                       o  u
Logical  Sym    T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev    E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   0060 RW   15509       0 RW 0060 WD   15425       0 A...  .  -   Consistent
DEV002   0061 RW    6737       0 RW 0061 WD    6669       0 A...  .  -   Consistent
DEV003   0062 RW   14986       0 RW 0062 WD   14909       0 A...  .  -   Consistent
DEV004   0063 RW    6607       0 RW 0063 WD    6521       0 A...  .  -   Consistent
DEV005   0064 RW    6293       0 RW 0064 WD    6231       0 A...  .  -   Consistent
DEV006   0065 RW    6840       0 RW 0065 WD    6765       0 A...  .  -   Consistent
DEV007   0066 RW    6884       0 RW 0066 WD    6811       0 A...  .  -   Consistent
DEV008   0067 RW   13244       0 RW 0067 WD   13152       0 A...  .  -   Consistent

Total            ------- -------            ------- -------
  Track(s)         77100       0              76483       0
  MBs             4818.8     0.0             4780.2     0.0
```

- The following query monitors the clearing of invalid tracks and reports on the query's progress every 60 minutes. The `-invalid` option verifies all invalid tracks on R1 and R2 are cleared before declaring the SRDF pairs in `HOP1` are in the Consistent with no invalid tracks state.

symrdf -cg HOP1 verify -consistent -noinvalids -i 60

```
Not All devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.

Not All devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.

All devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.
```

- The following query further confirms all `HOP1` device pairs are Consistent without any invalid tracks:

symrdf -cg HOP1 query

```
Composite Group Name      :  HOP1
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192600282    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603139    (Microcode Version: 5875)
RDF (RA) Group Number             :  98 (61)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
               ST                 LI      ST                       C  S
  Standard      A                  N       A                       o  u
Logical  Sym    T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev    E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   0060 RW       0       0 RW 0060 WD       0       0 A...  .  -   Consistent
DEV002   0061 RW       0       0 RW 0061 WD       0       0 A...  .  -   Consistent
DEV003   0062 RW       0       0 RW 0062 WD       0       0 A...  .  -   Consistent
DEV004   0063 RW       0       0 RW 0063 WD       0       0 A...  .  -   Consistent
DEV005   0064 RW       0       0 RW 0064 WD       0       0 A...  .  -   Consistent
DEV006   0065 RW       0       0 RW 0065 WD       0       0 A...  .  -   Consistent
DEV007   0066 RW       0       0 RW 0066 WD       0       0 A...  .  -   Consistent
DEV008   0067 RW       0       0 RW 0067 WD       0       0 A...  .  -   Consistent
```

```
Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0
```

- The following verify summary shows all `HOP1` device pairs are Consistent even though there are invalid tracks remaining to be resolved. The `-noinvalids` option prevents this command from succeeding until the invalid tracks are cleared.

symrdf verify -summary -consistent -noinvalids -cg HOP1 -i 45

```
Composite Group (CG) Name : HOP1

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        8
  Synchronized                      0
  SyncInProg                        0
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                             8

  RDF Mode                      Count
  --------------------------  ------
  Synchronous                       0
  Semi-Synchronous                  0
  Asynchronous                      8
  Adaptive Copy Write Pending       0
  Adaptive Copy Disk                0
  --------------------------  ------
  Total                             8

                                Track(s)    MB(s)
                               -----------  -------
  Total Source R1 Invalid          279611  17475.7
  Total Source R2 Invalid               0      0.0

  Total Target R1 Invalid          278653  17415.8
  Total Target R2 Invalid               0      0.0
```

None of the devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.

```
Composite Group (CG) Name : HOP1

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        8
  Synchronized                      0
  SyncInProg                        0
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                             8
```

```
  RDF Mode                       Count
  ---------------------------  ------
  Synchronous                       0
  Semi-Synchronous                  0
  Asynchronous                      8
  Adaptive Copy Write Pending       0
  Adaptive Copy Disk                0
  ---------------------------  ------
  Total                             8

                               Track(s)    MB(s)
                              -----------  -------
  Total Source R1 Invalid           53422   3338.9
  Total Source R2 Invalid               0      0.0

  Total Target R1 Invalid           52705   3294.1
  Total Target R2 Invalid               0      0.0

  Synchronization rate         :    314.2 MB/S
  Estimated time to completion : 00:00:11
```

Not All devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.

```
Composite Group (CG) Name : HOP1

  RDF Pair State                Count
  -----------------------      ------
  Consistent                        8
  Synchronized                      0
  SyncInProg                        0
  Suspended                         0
  Split                             0
  Failed Over                       0
  R1 Updated                        0
  R1 UpdInProg                      0
  TransIdle                         0
  Partitioned                       0
  Invalid                           0
  -----------------------      ------
  Total                             8

  RDF Mode                       Count
  ---------------------------  ------
  Synchronous                       0
  Semi-Synchronous                  0
  Asynchronous                      8
  Adaptive Copy Write Pending       0
  Adaptive Copy Disk                0
  ---------------------------  ------
  Total                             8

                               Track(s)    MB(s)
                              -----------  -------
  Total Source R1 Invalid               0      0.0
  Total Source R2 Invalid               0      0.0

  Total Target R1 Invalid               0      0.0
  Total Target R2 Invalid               0      0.0
```

All devices in the group 'HOP1' are in 'Consistent with no invalid tracks' state.

CHAPTER 15
Implementing Consistency Protection

This chapter provides SYMCLI examples for implementing consistency protection across one or more database management systems within an SRDF configuration using SRDF Enginuity Consistency Assist (RDF-ECA) for synchronous mode and SRDF Multi Session Consistency (MSC) for asynchronous mode:

- Example 1: Consistency protection in asynchronous mode 378
- Example 2: Tripping a consistency group automatically 385
- Example 3: Tripping a consistency group manually 388
- Example 4: Creating a composite group from existing sources 393
- Example 5: Consistency protection for concurrent SRDF..... 396
- Example 6: Dynamic modification of SRDF consistency groups 404
- Example 7: Recovering from a failed dynamic add operation 417

Note: Some of the examples in this section were performed with lower versions of software. Therefore, your output displays may not look exactly like the ones appearing in these examples.

Example 1: Consistency protection in asynchronous mode

This example was performed using Solutions Enabler version 6.1. A host is connected to local Symmetrix 000190300150, which is SRDF-connected to a remote Symmetrix array (000190300152). The SRDF daemon is installed on the host. The example uses the SRDF/A devices from local SRDF (RA) groups 25 and 26:

- The `symcfg list` command with the `-rdfg all` option displays a list of SRDF (RA) groups on the source Symmetrix array connected to the local host. All SRDF groups on a Symmetrix array are capable of SRDF/A operations. The RDFA "Flags C" column of RA groups 25 and 26 indicates N/A (-), which means these groups are not operating in async mode. The ellipsis (...) represents omitted output:

```
symcfg list -rdfg all -sid 150

Symmetrix ID : 000190300150

                S Y M M E T R I X   R D F   G R O U P S

    Local             Remote                    Group               RDFA Info
-------------  --------------------  -----------------------  ---------------
          LL                                     Flags  Dir  Flags  Cycle
RA-Grp  (sec)  RA-Grp  SymmID        T    Name     LPDS  Cfg   CSRM   time  Pri
-------------  --------------------  -----------------------  -----  -----  -
 1 ( 0)    10   1 ( 0) 000190300152  D bp4         XX..  F-S   XAM-   30    33
 2 ( 1)    10   2 ( 1) 000190300152  D power1      XX..  F-S   -IS-   30    33
 3 ( 2)    10   4 ( 3) 000190300152  D power2      XX..  F-S   -IS-   30    33
 4 ( 3)    10   9 ( 8) 000190300152  D power3      XX..  F-S   -IS-   30    33
 8 ( 7)    10   8 ( 7) 000190300152  D dav3        XX..  F-S   -IS-   30    33
10 ( 9)    10  10 ( 9) 000190300180  D test        XX..  F-S   -IS-   30    33
11 ( A)    10   3 ( 2) 000190300152  D bp1         XX..  F-S   XAM-   30    33
21 (14)    10  21 (14) 000190300180  D snhe2121    XX.N  F-S   -IS-   30    33
22 (15)    10  22 (15) 000190300152  D snhi2222    XX.N  F-S   .IS-   30    33
23 (16)    10  23 (16) 000190300180  D snhe2323    XX.N  F-S   -IS-   30    33
24 (17)    10  24 (17) 000190300152  D snhi2424    XX.N  F-S   .IS-   30    33
25 (18)    10  25 (18) 000190300152  D grp25       XX..  F-S   -IS-   30    33
26 (19)    10  26 (19) 000190300152  D grp26       XX..  F-S   -IS-   30    33
30 (1D)    10  30 (1D) 000190300152  D dav1        XX..  F-S   -IS-   30    33
.......................................................................................

Legend:
  ?                 : Unknown
  Group (T)ype      :  S = Static, D = Dynamic
  Director (C)onfig :  F-S = Fibre-Switched, F-H = Fibre-Hub
                       G = GIGE, E = ESCON, T = T3, - = N/A
  Group Flags       :
      Prevent Auto (L)ink Recovery      : X = Enabled, . = Disabled
      Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
      Link (D)omino                     : X = Enabled, . = Disabled
      (S)TAR mode                       : N = Normal, R = Recovery, . = OFF
  RDFA Flags        :
      (C)onsistency : X = Enabled, . = Disabled, - = N/A
      (S)tatus      : A = Active, I = Inactive, - = N/A
      (R)DFA Mode   : S = Single-session, M = MSC, - = N/A
      (M)sc Cleanup : C = MSC Cleanup required, - = N/A
```

- The `symcg create` command creates an RDF1-type composite group named `oracle`. You must specify the `-rdf_consistency` option to make the group capable of being enabled for SRDF consistency protection.

```
symcg create oracle -type rdf1 -rdf_consistency
```

- The `symcg addall` command adds standard devices from the source Symmetrix array (`-sid` 150) to the composite group, using the `-rdfg` option to add all devices from SRDF groups 25 and 26:

```
symcg -cg oracle -sid 150 -rdfg 25 addall dev
symcg -cg oracle -sid 150 -rdfg 26 addall dev
```

- The `symcg list` command displays a list of composite groups defined on this host. Only one composite group is defined, and it contains two devices, one in each SRDF (RAG) group. Include the `-rdf_consistency` option to display only those groups that are in the SRDF consistency database:

```
symcg list -rdf_consistency

                    C O M P O S I T E    G R O U P S

                                          Number of          Number of
    Name                  Type    Valid  Symms    RAGs    Devs    BCVs   VDEVs

    oracle                RDF1    Yes        1       2       2       0       0
```

- The `symrdf set mode async` command sets the method of replication to asynchronous for the devices in the composite group:

```
symrdf -cg oracle set mode async -noprompt

An RDF Set 'Asynchronous Mode' operation execution is in
progress for composite group 'oracle'. Please wait...

The RDF Set 'Asynchronous Mode' operation successfully executed
for composite group 'oracle'.
```

- The `symcg show` command displays configuration and status information about the composite group. The **A** and **S** entries in the **Flags S** and **R** columns of each device indicate that SRDF/A is now active but still operating in single-session mode (the Symmetrix controls SRDF/A session management). A period (.) in the **C** column means SRDF consistency is not yet enabled. When consistency is enabled, then the entire composite group will be enabled for consistency protection. If the links are up when the enable is performed, then the SRDF (RA) groups will go from single-session mode to MSC mode (the SRDF daemon controls RDFA session management). If the links are not up when the enable is performed, then the SRDF (RA) groups will go into MSC mode when the links are brought up through an operation such as `establish` or `resume`. SRDF Consistency Protection Allowed depends on your creating the composite group using the `-rdf_consistency` option:

```
symcg show oracle

Composite Group Name:  oracle

  Composite Group Type                              : RDF1
  Valid                                             : Yes
  CG in PowerPath                                   : No
  CG in GNS                                         : No
  RDF Consistency Protection Allowed                : Yes
  RDF Consistency Mode                              : NONE
  Concurrent RDF                                    : No

  Number of RDF (RA) Groups                         :    2
  Number of STD Devices                             :    2
  Number of CRDF STD Devices                        :    0
  Number of BCV's (Locally-associated)              :    0
  Number of VDEV's (Locally-associated)             :    0
  Number of RVDEV's (Remotely-associated VDEV)      :    0
  Number of RBCV's (Remotely-associated STD-RDF)    :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)   :    0
  Number of RRBCV's (Remotely-associated RBCV)      :    0

  Number of Symmetrix Units (1):
```

```
    {

    1) Symmetrix ID                                  : 000190300150
       Microcode Version                             : 5771
       Number of STD Devices                         :    2
       Number of CRDF STD Devices                    :    0
       Number of BCV's (Locally-associated)          :    0
       Number of VDEV's (Locally-associated)         :    0
       Number of RVDEV's (Remotely-associated VDEV)  :    0
       Number of RBCV's (Remotely-associated STD_RDF) :    0
       Number of BRBCV's (Remotely-associated BCV-RDF):    0
       Number of RRBCV's (Remotely-associated RBCV)  :    0

         Number of RDF (RA) Groups (2):
           {

           1) RDF (RA) Group Number  : 25             (18)
              Remote Symmetrix ID    : 000190300152
              Microcode Version      : 5771
              Recovery RA Group      : N/A             (N/A)
              RA Group Name          : N/A

              STD Devices (1):
               {
             ------------------------------------------------------------------------
                                          Sym  Device    Flags    Cap
   LdevName    PdevName                   Dev  Config    Sts      CSR     (MB)
             ------------------------------------------------------------------------
    DEV001      N/A                        0232 RDF1+R-5   RW     .AS     14370
             }

          2) RDF (RA) Group Number  : 26            (19)
             Remote Symmetrix ID    : 000190300152
             Microcode Version      : 5771
             Recovery RA Group      : N/A            (N/A)
             RA Group Name          : N/A

             STD Devices (1):
               {
             ------------------------------------------------------------------------
                                          Sym  Device     Flags   Cap
   LdevName    PdevName                   Dev  Config     Sts     CSR     (MB)
             ------------------------------------------------------------------------
    DEV002      N/A                        0242 RDF1+R-5   RW     .AS     14370
             }
         }
     }

Legend:
  RDFA Flags:
       C(onsistency)   : X = Enabled, . = Disabled, - = N/A
       (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
       R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
```

- The `symcg set -name` commands are optional with a composite group that contains all asynchronous devices (as group `oracle` does) or all synchronous devices. If the composite group contains both asynchronous and synchronous devices and you wish to enable for consistency protection, then you must use the `symcg set -name` option. This has more relevance with concurrent SRDF when you want to control asynchronous SRDF groups separately from synchronous SRDF groups (refer to "Example 5: Consistency protection for concurrent SRDF" on page 396).

  Setting a name such as `oracleAsync` for the two SRDF groups (25 and 26) allows you to perform SRDF control operations on these SRDF groups using this name This example performs control operations on the composite group, not on the SRDF group name:

  `symcg -cg oracle set -name oracleAsync -rdfg 150:25,26`

- The `symrdf split` command splits the devices in the composite group named `oracle`:

`symrdf -cg oracle split -noprompt`

```
An RDF 'Split' operation execution is
in progress for composite group 'oracle'. Please wait...
```

```
    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.
    Read/Write Enable device(s) in (0150,25) on RA at target (R2)...Done.
    Read/Write Enable device(s) in (0150,26) on RA at target (R2)...Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.

The RDF 'Split' operation successfully executed for
composite group 'oracle'.
```

- The `symrdf query` command displays the state of the SRDF/A pairs for each SRDF group in the composite group `oracle`. Each SRDF pair is in the `Split` state. Including the `-detail` option provides the most information:

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names:
    {
    RDFG Name                          : oracleAsync
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 26 (19) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 0
    Session Status                     : Inactive
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:00
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:00
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------ ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ----------
DEV002   0242 RW       0  294766 NR 0242 RW       0       0 A...   Split

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 25 (18) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 0
    Session Status                     : Inactive
```

```
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:00
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:00
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV001   0232 RW       0  294784 NR 0232 RW       0       0 A...   Split

Total            ------- -------            ------- -------
  Track(s)             0  589550                  0       0
  MBs                0.0 18423.4                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The following command initiates the synchronization of SRDF pairs in the composite group:

symrdf -cg oracle establish -noprompt

```
An RDF 'Incremental Establish' operation execution is
in progress for composite group 'oracle'. Please wait...

    Write Disable device(s) in (0150,26) on RA at target (R2).......Done.
    Write Disable device(s) in (0150,25) on RA at target (R2).......Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.
    Resume RDF link(s) for device(s) in (0150,26)...................Started.
    Resume RDF link(s) for device(s) in (0150,25)...................Started.
    Merge track tables between source and target in (0150,25).......Started.
    Devices: 0232-0241 ........................................... Merged.
    Merge track tables between source and target in (0150,26).......Started.
    Devices: 0242-0251 ........................................... Merged.
    Merge track tables between source and target in (0150,25).......Done.
    Merge track tables between source and target in (0150,26).......Done.
    Resume RDF link(s) for device(s) in (0150,25)...................Done.
    Resume RDF link(s) for device(s) in (0150,26)...................Done.

The RDF 'Incremental Establish' operation successfully initiated for
composite group 'oracle'.
```

- All pairs are in the process of becoming consistent (SyncInProg) and moving toward the Consistent state ("Consistent" is a state characteristic of SRDF/A pairs). The `symrdf verify` command checks the state of SRDF/A pairs in the composite group every 60 seconds until all pairs are in the Consistent state:

```
symrdf -cg oracle -consistent verify -i 60

None of the devices in the group 'oracle' are in 'Consistent' state.

None of the devices in the group 'oracle' are in 'Consistent' state.
.....................................................................
Not all of the devices in the group 'oracle' are in 'Consistent' state.

All devices in the group 'oracle' are in 'Consistent' state.
```

- The `symcg enable` command enables SRDF consistency protection for the composite group. The group is now known as a consistency group. At this point, the SRDF daemon takes over RDFA session management from the Symmetrix array:

```
symcg -cg oracle enable -noprompt

A consistency 'Enable' operation execution is
in progress for composite group 'oracle'. Please wait...

The consistency 'Enable' operation successfully executed for
composite group 'oracle'.
```

- The `symrdf verify` command with the `-cg_consistent` option checks every 120 seconds to determine when the consistency group reaches the Consistent state. This state occurs when at least two cycle switches have occurred since all devices in each SRDF (RA) group reached a consistent state:

```
symrdf -cg oracle verify -cg_consistent -i 120

CG 'oracle' is NOT RDF-Consistent.
CG 'oracle' is RDF-Consistent.
```

- The symrdf query command with the -detail option shows that the consistency protection for the group is enabled and all SRDF/A device pairs are in the Consistent state. RDFA MSC Consistency Info shows that the Consistency State of the consistency group is also CONSISTENT, meaning that some cycle switches have occurred since all devices in each SRDF (RA) group reached the Consistent state. Note that the RDFG Names information is relevant only if you enable SRDF consistency using the RDFG Name instead of using the Composite group name:

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info:
    {
    Session Status                     : Active
    Consistency State                  : CONSISTENT
    }

RDFG Names:
    {
    RDFG Name                          : oracleAsync
```

```
    RDF Consistency Mode               : NONE
    }


Symmetrix ID                       : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID                : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number              : 26 (19) - oracleAsync
Star Mode                          : NO
RDFA Info:
    {
    Cycle Number                       : 8
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:31
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:56
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }


        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0242 RW       0       0 RW 0242 WD       0       0 A..X   Consistent

Symmetrix ID                       : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID                : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number              : 25 (18) - oracleAsync
Star Mode                          : NO
RDFA Info:
    {
    Cycle Number                       : 8
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:31
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:56
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV001   0232 RW       0       0 RW 0232 WD       0       0 A..X   Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
```

```
 D(omino)            : X = Enabled, . = Disabled
 A(daptive Copy)     : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

Example 2: Tripping a consistency group automatically

This example is a continuation of Example 1. The link represented by SRDF (RA) group 26 is disconnected to simulate an automatic trip (unplanned interruption) of the consistency group. I/O continues to occur on the local Symmetrix array:

- At this point, the link represented by SRDF (RA) group 26 is "disconnected." The query checks the status of SRDF/A pairs. The Partitioned state indicates that a physical link is down between an R1 device and its R2 target device. If only one link goes down and the other stays up, the latter goes to the Suspended state, and consistency on that link is maintained. The SRDF daemon recognizes the interruption and suspends the other SRDF link in the consistency group (SRDF group 25) in a manner that honors dependent write I/Os. Recall that consistency protection is suspended when one or more R1 devices in a consistency group cannot propagate data to their corresponding R2 target devices. Although consistency protection is temporarily suspended, "SRDF consistency" remains enabled:

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info
    {
    Session Status                     : Inactive
    Consistency State                  : N/A
    }

RDFG Names
    {
    RDFG Name                          : oracleAsync
    RDF Consistency Mode               : NONE
    }


Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : N/A             (Microcode Version: N/A)
RDF (RA) Group Number             : 26 (19) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 0
    Session Status                     : Inactive
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:00
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:00
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }


        Source (R1) View             Target (R2) View       MODES
```

```
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0242 RW       0   35094 NR 0242 NA      NA      NA A..X   Partitioned

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 25 (18) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                      : 0
    Session Status                    : Inactive
    Minimum Cycle Time                : 00:00:30
    Avg Cycle Time                    : 00:00:30
    Duration of Last cycle            : 00:00:30
    Session Priority                  : 33
    Tracks not Committed to the R2 Side: 39484
    Time that R2 is behind R1         : 00:02:24
    R1 Side Percent Cache In Use      :  4
    R2 Side Percent Cache In Use      :  8
    }

        Source (R1) View            Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV001   0232 RW       0       0 RW 0232 WD       0       0 A..X   Suspended

Total            ------- -------            ------- -------
  Track(s)             0   35094                  0       0
  MBs                0.0  1096.7                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- At this point, the RA link is reconnected. Once the link is repaired, the SRDF pair state changes from Partitioned to Suspended. The `symrdf query` command displays this revised state.

symrdf -cg oracle query -detail

```
Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info
    {
    Session Status                    : Inactive
    Consistency State                 : N/A
    }

RDFG Names
    {
    RDFG Name                         : oracleAsync
```

```
    RDF Consistency Mode                : NONE
    }


Symmetrix ID                        : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID                 : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number               : 26 (19) - oracleAsync
Star Mode                           : NO
RDFA Info:
    {
    Cycle Number                        : 0
    Session Status                      : Inactive
    Minimum Cycle Time                  : 00:00:30
    Avg Cycle Time                      : 00:00:00
    Duration of Last cycle              : 00:00:00
    Session Priority                    : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1           : 00:00:00
    R1 Side Percent Cache In Use        :  0
    R2 Side Percent Cache In Use        :  0
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0242 RW       0   35094 NR 0242 WD    4326       0 A..X   Suspended

Symmetrix ID                        : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID                 : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number               : 25 (18) - oracleAsync
Star Mode                           : NO

RDFA Info:
    {
    Cycle Number                        : 0
    Session Status                      : Inactive - MSC
    Minimum Cycle Time                  : 00:00:30
    Avg Cycle Time                      : 00:00:00
    Duration of Last cycle              : 00:00:00
    Session Priority                    : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1           : 00:00:00
    R1 Side Percent Cache In Use        :  1
    R2 Side Percent Cache In Use        :  4
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV001   0232 RW       0    8110 NR 0232 WD   10862       0 A..X   Suspended

Total            ------- -------            ------- -------
  Track(s)             0   43204              15188       0
  MBs                0.0  1350.1              474.6     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
```

```
 D(omino)            : X = Enabled, . = Disabled
 A(daptive Copy)     : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The SRDF/A pairs remain in the Suspended state until you manually re-establish them. The `establish` action first initiates any cache cleanup that may have been needed because the physical links went down, resulting in the last cycle being committed or discarded from the cache. Then the SRDF/A session and consistency protection are automatically resumed:

```
symrdf -cg oracle establish -noprompt

An RDF 'Incremental Establish' operation execution is
in progress for composite group 'oracle'. Please wait...

    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Started.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.
    Mark target device(s) in (0150,25) to refresh from source.......Started.
    Devices: 0232-0239 ............................................ Marked.
    Devices: 023A-0241 ............................................ Marked.
    Mark target device(s) in (0150,26) to refresh from source.......Started.
    Devices: 024E-0250 ............................................ Marked.
    Mark target device(s) in (0150,26) to refresh from source.......Done.
    Mark target device(s) in (0150,25) to refresh from source.......Done.
    Merge track tables between source and target in (0150,25).......Started.
    Merge track tables between source and target in (0150,26).......Started.
    Devices: 024E-0250 ............................................ Merged.
    Merge track tables between source and target in (0150,26).......Done.
    Devices: 0232-0241 ............................................ Merged.
    Merge track tables between source and target in (0150,25).......Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.
    Resume RDF link(s) for device(s) in (0150,26)...................Started.
    Resume RDF link(s) for device(s) in (0150,25)...................Started.
    Resume RDF link(s) for device(s) in (0150,25)...................Done.
    Resume RDF link(s) for device(s) in (0150,26)...................Done.

The RDF 'Incremental Establish' operation successfully initiated for
composite group 'oracle'.
```

- The `symrdf verify` command with the `-cg_consistent` option checks every 120 seconds to determine when the consistency group reaches the Consistent state:

```
symrdf -cg oracle verify -cg_consistent -i 120

CG 'oracle' is NOT RDF-Consistent.

CG 'oracle' is NOT RDF-Consistent.

CG 'oracle' is NOT RDF-Consistent.

CG 'oracle' is NOT RDF-Consistent.

CG 'oracle' is RDF-Consistent.
```

Example 3: Tripping a consistency group manually

This example continues from the end of Example 2 to determine if tripping the consistency group manually produces similar results to suspending consistency protection when an unplanned interruption occurs:

- The `symrdf suspend` command deactivates the consistency group. The -force parameter confirms that you really want to stop the SRDF mirroring operation and suspend consistency protection:

```
symrdf  -cg oracle suspend -force

An RDF 'Suspend' operation execution is
in progress for composite group 'oracle'. Please wait...

    Suspend RDF link(s) for device(s) in (0150,26)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Started.
    Suspend RDF link(s) for device(s) in (0150,25)..................Done.
    Suspend RDF link(s) for device(s) in (0150,26)..................Done.

The RDF 'Suspend' operation successfully executed for
composite group 'oracle'.
```

- The following query with the -detail option shows that all R1 devices from the consistency group are in the Suspended state. Consistency protection is temporarily Inactive but remains enabled:

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info
    {
    Session Status                     : Inactive
    Consistency State                  : N/A
    }

RDFG Names
    {
    RDFG Name                          : oracleAsync
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 26 (19) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 0
    Session Status                     : Inactive
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:00
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:00
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View            Target (R2) View       MODES
-------------------------------  ------------------------ ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ----------
```

```
DEV002   0242 RW       0   74806 NR 0242 WD       0       0 A..X   Suspended

Symmetrix ID                     : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID              : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number            : 25 (18) - oracleAsync
Star Mode                        : NO
RDFA Info:
    {
    Cycle Number                      : 0
    Session Status                    : Inactive
    Minimum Cycle Time                : 00:00:30
    Avg Cycle Time                    : 00:00:00
    Duration of Last cycle            : 00:00:00
    Session Priority                  : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1         : 00:00:00
    R1 Side Percent Cache In Use      :  0
    R2 Side Percent Cache In Use      :  0
    }

        Source (R1) View            Target (R2) View       MODES
-------------------------------  ------------------------- ----- ----------
              ST                LI      ST
  Standard     A                 N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------- ----- ----------
DEV001   0232 RW       0   74824 NR 0232 WD       0       0 A..X   Suspended

Total            ------- -------            ------- -------
  Track(s)             0  149630                  0       0
  MBs                0.0  4675.9                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The `symrdf resume` command resumes the SRDF links between the SRDF/A pairs in the consistency group and I/O traffic between the R1 devices and their paired R2 devices. Normal SRDF mirroring resumes. Consistency protection is automatically activated again upon resumption of the link:

```
symrdf -cg oracle resume

An RDF 'Resume' operation execution is
in progress for composite group 'oracle'. Please wait...

    Resume RDF link(s) for device(s) in (0150,26)...................Started.
    Resume RDF link(s) for device(s) in (0150,25)...................Started.
    Resume RDF link(s) for device(s) in (0150,25)...................Done.
    Resume RDF link(s) for device(s) in (0150,26)...................Done.

The RDF 'Resume' operation successfully executed for
composite group 'oracle'.
```

- The `symrdf query` command displays the state of the SRDF/A pairs, which are in the process of becoming consistent (the **SyncInProg** state to the Consistent state):

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
```

```
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info
    {
    Session Status                     : Active
    Consistency State                  : INCONSISTENT
    }

RDFG Names
    {
    RDFG Name                          : oracleAsync
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 26 (19) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 2
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:28
    Duration of Last cycle             : 00:00:28
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 17274
    Time that R2 is behind R1          : 00:00:53
    R1 Side Percent Cache In Use       :  3
    R2 Side Percent Cache In Use       :  2
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0242 RW       0  143372 RW 0242 WD       0       0 A..X   SyncInProg

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 25 (18) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 2
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:30
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 14882
    Time that R2 is behind R1          : 00:00:55
    R1 Side Percent Cache In Use       :  2
    R2 Side Percent Cache In Use       :  3
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
```

```
------------------------------- -- ----------------------- ----- ----------
DEV001   0232 RW       0  147046 RW 0232 WD       0       0 A..X   SyncInProg

Total            ------- -------            ------- -------
  Track(s)             0  290418                  0       0
  MBs                0.0  9075.6                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- A subsequent `symrdf query` command shows that all SRDF device pairs are now in the **Consistent** state and that the consistency group is **CONSISTENT** again:

```
symrdf -cg oracle query -detail

Composite Group Name      :  oracle
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  MSC
RDFA MSC Consistency Info
    {
    Session Status                     : Active
    Consistency State                  : CONSISTENT
    }

RDFG Names
    {
    RDFG Name                          : oracleAsync
    RDF Consistency Mode               : NONE
    }


Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             : 26 (19) - oracleAsync
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 38
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:30
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:40
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
------------------------------- -- ----------------------- ----- ----------
DEV002   0242 RW       0       0 RW 0242 WD       0       0 A..X   Consistent

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
```

```
RDF (RA) Group Number             : 25 (18) - oracleAsync
Star Mode                         : NO

RDFA Info:
    {
    Cycle Number                       : 38
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:30
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:40
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------ ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ----------
DEV001   0232 RW       0       0 RW 0232 WD       0       0 A..X   Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

Example 4: Creating a composite group from existing sources

This example populates a composite group using devices from an existing device group.

- The `symdg list` command displays two device groups (`ora1` and `ora2`) containing SRDF devices that can be included in a composite group:

```
symdg list

                         D E V I C E     G R O U P S

                                                          Number of
    Name               Type     Valid  Symmetrix ID  Devs   GKs  BCVs  VDEVs

    ora1               RDF1     Yes    000187400011    16     0     0      0
    ora2               RDF1     Yes    000187400011    16     0     0      0
```

- The `symdg dg2cg` command creates and populates a composite group named `oracle`, using devices from a device group named `ora1`. The `-rdf_consistency` option creates the composite group in the host's RDF consistency database:

```
symdg dg2cg ora1 oracle -rdf_consistency

Adding STD device 0FB2 to group 'oracle'... OK
Adding STD device 0FB3 to group 'oracle'... OK
Adding STD device 0FB4 to group 'oracle'... OK
Adding STD device 0FB5 to group 'oracle'... OK
```

```
Adding STD device 0FB6 to group 'oracle'... OK
Adding STD device 0FB7 to group 'oracle'... OK
Adding STD device 0FB8 to group 'oracle'... OK
Adding STD device 0FB9 to group 'oracle'... OK
Adding STD device 0FBA to group 'oracle'... OK
Adding STD device 0FBB to group 'oracle'... OK
Adding STD device 0FBC to group 'oracle'... OK
Adding STD device 0FBD to group 'oracle'... OK
Adding STD device 0FBE to group 'oracle'... OK
Adding STD device 0FBF to group 'oracle'... OK
Adding STD device 0FC0 to group 'oracle'... OK
Adding STD device 0FC1 to group 'oracle'... OK

16 device(s) were added to group 'oracle'.
```

- The `symcg list` command displays a list of composite groups defined on this host. This display shows the new composite group and that 16 devices were added to it from the device group:

symcg list

```
                    C O M P O S I T E   G R O U P S

                                        Number of          Number of
    Name                  Type    Valid  Symms   RAGs    Devs   BCVs   VDEVs

    oracle                RDF1    Yes        1      1      16      0       0
```

- This `symdg dg2cg` command adds devices from `ora2` to the same composite group. You need to use the `-rename` option with this second `dg2cg` command because logical device names are carried from the device group to the composite group. In the case of default logical device names, those from `ora2` can collide with the same logical device names from `ora1`. The `-rename` option generates new logical device names for the devices being added from a second device group:

```
symdg dg2cg ora2 oracle -rdf_consistency -rename

Adding STD device 0FC2 to group 'oracle'... OK
Adding STD device 0FC3 to group 'oracle'... OK
Adding STD device 0FC4 to group 'oracle'... OK
Adding STD device 0FC5 to group 'oracle'... OK
Adding STD device 0FC6 to group 'oracle'... OK
Adding STD device 0FC7 to group 'oracle'... OK
Adding STD device 0FC8 to group 'oracle'... OK
Adding STD device 0FC9 to group 'oracle'... OK
Adding STD device 0FCA to group 'oracle'... OK
Adding STD device 0FCB to group 'oracle'... OK
Adding STD device 0FCC to group 'oracle'... OK
Adding STD device 0FCD to group 'oracle'... OK
Adding STD device 0FCE to group 'oracle'... OK
Adding STD device 0FCF to group 'oracle'... OK
Adding STD device 0FD0 to group 'oracle'... OK
Adding STD device 0FD1 to group 'oracle'... OK

16 device(s) were added to group 'oracle'.
```

- This `symcg list` command shows that the `oracle` composite group now contains the 32 devices (16 from each device group):

symcg list

```
                    C O M P O S I T E   G R O U P S

                                        Number of          Number of
```

```
Name                    Type    Valid  Symms   RAGs    Devs   BCVs   VDEVs

oracle                  RDF1    Yes        1      2      32      0       0
```

- The `symcg show` command displays the details of the composite group `oracle`. Note that the logical device names of the `ora2` devices (the second set of sixteen) were renamed in the composite group to be DEV017 through DEV032:

```
symcg show oracle

Composite Group Name:  oracle

  Composite Group Type                                : RDF1
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : No
  RDF Consistency Protection Allowed                  : Yes
  RDF Consistency Enabled                             : No

  Number of RDF (RA) Groups                           :    2
  Number of STD Devices                               :   32
  Number of BCV's (Locally-associated)                :    0
  Number of VDEV's (Locally-associated)               :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)     :    0
  Number of RRBCV's (Remotely-associated RBCV)        :    0

  Number of Symmetrix Units (1):
    {

    1) Symmetrix ID                                    : 000187400011
       Microcode Version                               : 5671
       Number of STD Devices                           :   32
       Number of BCV's (Locally-associated)            :    0
       Number of VDEV's (Locally-associated)           :    0
       Number of RBCV's (Remotely-associated STD_RDF)  :    0
       Number of BRBCV's (Remotely-associated BCV-RDF):    0
       Number of RRBCV's (Remotely-associated RBCV)    :    0

       Number of RDF (RA) Groups (2):
         {

         1) RDF (RA) Group Number  : 27            (1A)
            Remote Symmetrix ID    : 000187400093
            Microcode Version      : 5671


            STD Devices (16):
              {

   ------------------------------------------------------------------------
                                        Sym  Device     Flags   Cap
    LdevName    PdevName                Dev  Config     Sts     CSR     (MB)

   ------------------------------------------------------------------------
    DEV001      /dev/rdsk/c68t12d0      0FB2 RDF1         RW    .AS     449
    DEV002      /dev/rdsk/c68t12d1      0FB3 RDF1         RW    .AS     449
    DEV003      /dev/rdsk/c68t12d2      0FB4 RDF1         RW    .AS     449
    DEV004      /dev/rdsk/c68t12d3      0FB5 RDF1         RW    .AS     449
    DEV005      /dev/rdsk/c68t12d4      0FB6 RDF1         RW    .AS     449
    DEV006      /dev/rdsk/c68t12d5      0FB7 RDF1         RW    .AS     449
    DEV007      /dev/rdsk/c68t12d6      0FB8 RDF1         RW    .AS     449
    DEV008      /dev/rdsk/c68t12d7      0FB9 RDF1         RW    .AS     449
    DEV009      /dev/rdsk/c68t13d0      0FBA RDF1         RW    .AS     449
    DEV010      /dev/rdsk/c68t13d1      0FBB RDF1         RW    .AS     449
    DEV011      /dev/rdsk/c68t13d2      0FBC RDF1         RW    .AS     449
```

```
    DEV012      /dev/rdsk/c68t13d3       0FBD RDF1         RW    .AS     449
    DEV013      /dev/rdsk/c68t13d4       0FBE RDF1         RW    .AS     449
    DEV014      /dev/rdsk/c68t13d5       0FBF RDF1         RW    .AS     449
    DEV015      /dev/rdsk/c68t13d6       0FC0 RDF1         RW    .AS     449
    DEV016      /dev/rdsk/c68t13d7       0FC1 RDF1         RW    .AS     449
              }

         2) RDF (RA) Group Number  : 28            (1B)
            Remote Symmetrix ID    : 000187400093
            Microcode Version      : 5671

            STD Devices (16):
              {

   ---------------------------------------------------------------------------
                                        Sym Device       Flags   Cap
    LdevName    PdevName                Dev Config       Sts     CSR    (MB)

   ---------------------------------------------------------------------------
    DEV017      /dev/rdsk/c68t14d0       0FC2 RDF1        RW    .AS       449
    DEV018      /dev/rdsk/c68t14d1       0FC3 RDF1        RW    .AS       449
    DEV019      /dev/rdsk/c68t14d2       0FC4 RDF1        RW    .AS       449
    DEV020      /dev/rdsk/c68t14d3       0FC5 RDF1        RW    .AS       449
    DEV021      /dev/rdsk/c68t14d4       0FC6 RDF1        RW    .AS       449
    DEV022      /dev/rdsk/c68t14d5       0FC7 RDF1        RW    .AS       449
    DEV023      /dev/rdsk/c68t14d6       0FC8 RDF1        RW    .AS       449
    DEV024      /dev/rdsk/c68t14d7       0FC9 RDF1        RW    .AS       449
    DEV025      /dev/rdsk/c68t15d0       0FCA RDF1        RW    .AS       449
    DEV026      /dev/rdsk/c68t15d1       0FCB RDF1        RW    .AS       449
    DEV027      /dev/rdsk/c68t15d2       0FCC RDF1        RW    .AS       449
    DEV028      /dev/rdsk/c68t15d3       0FCD RDF1        RW    .AS       449
    DEV029      /dev/rdsk/c68t15d4       0FCE RDF1        RW    .AS       449
    DEV030      /dev/rdsk/c68t15d5       0FCF RDF1        RW    .AS       449
    DEV031      /dev/rdsk/c68t15d6       0FD0 RDF1        RW    .AS       449
    DEV032      /dev/rdsk/c68t15d7       0FD1 RDF1        RW    .AS       449
              }
         }
      }

Legend:
  RDFA Flags:
       C(onsistency)   : X = Enabled, . = Disabled, - = N/A
       (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
       R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
```

Example 5: Consistency protection for concurrent SRDF

This example is performed using Solutions Enabler version 6.1. The hardware configuration for the following concurrent SRDF example consists of:

- Local Source Symmetrix (sid 150): R1 concurrent devices
- Remote Target Symmetrix (sid 180): R2 devices in synchronous mode
- Remote Target Symmetrix (sid 152): R2 devices to be run in asynchronous mode
- The symrdf list command with the -concurrent option shows devices on the local Symmetrix (sid 150) that are configured as concurrent SRDF devices. Each of two remote devices of a concurrent R1 device belongs to a different SRDF group (for example, SRDF Typ:G 4 and 5). Device 0072 is the meta head of a 16-member meta device (0072 to 0081), and device 0082 is the meta head of a second meta device. These two meta head devices display the invalid tracks for all members of the meta device:

symrdf list -sid 150 -concurrent

```
Symmetrix ID: 000190300150


                                Local Device View
-----------------------------------------------------------------------------
                     STATUS   MODES                          RDF  S T A T E S
Sym        RDF      --------- -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G    SA RA LNK  MDA   Tracks   Tracks Dev RDev Pair
---- ----  ------   --------- ----- -------  ------- --- ---- -------------

0072 0072   R1:4    RW RW RW   S..        0        0 RW  WD   Synchronized
     0072   R1:5    RW RW RW   S..        0        0 RW  WD   Synchronized
0073 0073   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0073   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0074 0074   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0074   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0075 0075   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0075   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0076 0076   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0076   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0077 0077   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0077   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0078 0078   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0078   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0079 0079   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0079   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007A 007A   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007A   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007B 007B   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007B   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007C 007C   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007C   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007D 007D   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007D   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007E 007E   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007E   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
007F 007F   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     007F   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0080 0080   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0080   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0081 0081   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0081   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0082 0082   R1:4    RW RW RW   S..        0        0 RW  WD   Synchronized
     0082   R1:5    RW RW RW   S..        0        0 RW  WD   Synchronized
0083 0083   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0083   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0084 0084   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0084   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0085 0085   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0085   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0086 0086   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0086   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0087 0087   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0087   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0088 0088   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0088   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
0089 0089   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     0089   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
008A 008A   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     008A   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
008B 008B   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     008B   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
008C 008C   R1:4    RW RW RW   S..        -        - RW  WD   Synchronized
     008C   R1:5    RW RW RW   S..        -        - RW  WD   Synchronized
```

```
008D 008D   R1:4  RW RW RW   S..        -        - RW  WD   Synchronized
     008D   R1:5  RW RW RW   S..        -        - RW  WD   Synchronized
008E 008E   R1:4  RW RW RW   S..        -        - RW  WD   Synchronized
     008E   R1:5  RW RW RW   S..        -        - RW  WD   Synchronized
008F 008F   R1:4  RW RW RW   S..        -        - RW  WD   Synchronized
     008F   R1:5  RW RW RW   S..        -        - RW  WD   Synchronized
0090 0090   R1:4  RW RW RW   S..        -        - RW  WD   Synchronized
     0090   R1:5  RW RW RW   S..        -        - RW  WD   Synchronized
0091 0091   R1:4  RW RW RW   S..        -        - RW  WD   Synchronized
     0091   R1:5  RW RW RW   S..        -        - RW  WD   Synchronized

Total                            -------- --------
  Track(s)                              0        0
  MB(s)                               0.0      0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
```

- The following command creates a composite group named `srdftest2`. The `-type` parameter specifies an RDF1 type group (for the R1 devices). The `-rdf_consistency` parameter indicates that the composite group will be added to the RDF consistency database so that it can be managed for SRDF consistency protection:

  symcg create srdftest2 -type rdf1 -rdf_consistency

- The following command adds to the composite group all devices from the two SRDF groups that represent the concurrent links. With concurrent R1 devices, the command that adds one concurrent link (for example, SRDF group 4) actually adds both concurrent links: SRDF group 4 and SRDF group 5:

  symcg -cg srdftest2 -sid 150 addall dev -rdfg 4

- The symrdf query command displays local and remote Symmetrix array information and the status of the SRDF pairs in the composite group. Currently, the SRDF pairs are in the Synchronized state. Both SRDF groups are initially operating in Sync mode (S); SRDF group 5 will be switched to Async mode later. You can use the -detail option to provide more information than the standard query operation. This is particularly useful if SRDF group names have been set and when RDFA information is needed (as is shown later):

symrdf -cg srdftest2 query -detail

```
Composite Group Name      :  srdftest2
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300180    (Microcode Version: 5771)
RDF (RA) Group Number             :  4 (03)
Star Mode                         : NO

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------- ----- ----------
```

```
DEV001   0072 RW       0       0 RW 0072 WD       0       0 S...   Synchronized
DEV002   0082 RW       0       0 RW 0082 WD       0       0 S...   Synchronized

Symmetrix ID                     : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID              : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number            :  5 (04)
Star Mode                        : NO

        Source (R1) View             Target (R2) View       MODES
-------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------- ----- ----------
DEV001   0072 RW       0       0 RW 0072 WD       0       0 S...   Synchronized
DEV002   0082 RW       0       0 RW 0082 WD       0       0 S...   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The `symcg set -name` commands create names for each SRDF group for use in controlling consistency with this name. For example, the names `rdfg4` for SRDF group 4, and `rdfg5` for SRDF group 5. Specify the `-rdfg` parameter and the `sid:rdfg` format (`150:4` means Symmetrix 150 and SRDF group 4):

  ```
  symcg -cg srdftest2 set -name rdfg4 -rdfg 150:4
  ```

  ```
  symcg -cg srdftest2 set -name rdfg5 -rdfg 150:5
  ```

- The `symrdf set mode async` command sets the method of replication to asynchronous (ASYNC) for the devices in the composite group subset named `rdfg5`. This begins asynchronous replication to Symmetrix 152, while continuing synchronous replication to Symmetrix 180:

  ```
  symrdf -cg srdftest2 -rdfg name:rdfg5 set mode async -noprompt
  ```

  ```
  An RDF Set 'Asynchronous Mode' operation execution is in
  progress for composite group 'srdftest2'. Please wait...

  The RDF Set 'Asynchronous Mode' operation successfully executed
  for composite group 'srdftest2'.
  ```

- Another symrdf query command shows that the SRDF pairs in rdfg5 (SRDF group 5, Symmetrix 152) are now in Async mode (A), and their pair state is Consistent. Note that the -detail option provides RDFG Names information and RDFA Info details:

```
symrdf -cg srdftest2 query -detail

Composite Group Name      :  srdftest2
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names
    {
```

```
    RDFG Name                          : rdfg4
    RDF Consistency Mode               : NONE

    RDFG Name                          : rdfg5
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300180    (Microcode Version: 5771)
RDF (RA) Group Number             :  4 (03) - rdfg4
Star Mode                         : NO

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ----------
DEV002   0082 RW       0       0 RW 0082 WD       0       0 S...   Synchronized
DEV001   0072 RW       0       0 RW 0072 WD       0       0 S...   Synchronized

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             :  5 (04) - rdfg5
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 1
    Session Status                     : Active
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:30
    Duration of Last cycle             : 00:00:00
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:21
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ------------------------ ----- ----------
DEV002   0082 RW       0       0 RW 0082 WD       0       0 A...   Consistent
DEV001   0072 RW       0       0 RW 0072 WD       0       0 A...   Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The `symcg enable` command enables SRDF consistency protection for the subset `rdfg4`:

```
symcg -cg srdftest2 -rdfg name:rdfg4 enable -noprompt
```

```
A consistency 'Enable' operation execution is
in progress for composite group 'srdftest2'. Please wait...

The consistency 'Enable' operation successfully executed for
composite group 'srdftest2'.
```

- Another `symcg enable` command enables SRDF consistency protection for the subset `rdfg5`:

```
symcg -cg srdftest2 -rdfg name:rdfg5 enable -noprompt

A consistency 'Enable' operation execution is
in progress for composite group 'srdftest2'. Please wait...

The consistency 'Enable' operation successfully executed for
composite group 'srdftest2'.
```

- Tripping a subset of the consistency group manually produces similar results to suspending consistency protection when an unplanned interruption occurs. The following `symrdf suspend` command deactivates the `rdfg4` subset, thus suspending the synchronous link. The `-force` option is required here to ensure that you really want to stop the SRDF mirroring operation and suspend consistency protection on the synchronous link:

```
symrdf -cg srdftest2 suspend -rdfg name:rdfg4 -noprompt -force

An RDF 'Suspend' operation execution is
in progress for composite group 'srdftest2'. Please wait...

    Pend I/O on RDF link(s) for device(s) in (0150,04)..............Done.
    Suspend RDF link(s) for device(s) in (0150,04)..................Done.

The RDF 'Suspend' operation successfully executed for
composite group 'srdftest2'.
```

- Another `symrdf query` command shows that the synchronous link is **Suspended** but the asynchronous link is unaffected:

```
symrdf -cg srdftest2 query -detail

Composite Group Name      :  srdftest2
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names
    {
    RDFG Name                          : rdfg4
    RDF Consistency Mode               : SYNC
    Sync Consistency Info
        {
        Consistency State              : N/A
        }

    RDFG Name                          : rdfg5
    RDF Consistency Mode               : MSC
    MSC Consistency Info
        {
        Session Status                 : Active
        Consistency State              : Consistent
        }
    }
```

```
Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300180    (Microcode Version: 5771)
RDF (RA) Group Number             :  4 (03) - rdfg4
Star Mode                         : NO

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0082 RW       0       0 NR 0082 WD       0       0 S..X   Suspended
DEV001   0072 RW       0       0 NR 0072 WD       0       0 S..X   Suspended

Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             :  5 (04) - rdfg5
Star Mode                         : NO
RDFA Info:
    {
    Cycle Number                       : 39
    Session Status                     : Active - MSC
    Minimum Cycle Time                 : 00:00:30
    Avg Cycle Time                     : 00:00:30
    Duration of Last cycle             : 00:00:30
    Session Priority                   : 33
    Tracks not Committed to the R2 Side: 0
    Time that R2 is behind R1          : 00:00:51
    R1 Side Percent Cache In Use       :  0
    R2 Side Percent Cache In Use       :  0
    }

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ----------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAC  STATE
-------------------------------- -- ----------------------- ----- ----------
DEV002   0082 RW       0       0 RW 0082 WD       0       0 A..X   Consistent
DEV001   0072 RW       0       0 RW 0072 WD       0       0 A..X   Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State): X = Enabled, . = Disabled, - = N/A
```

- The `symrdf resume` command resumes the synchronous SRDF link represented by the `rdfg4` subset of the consistency group. I/O traffic resumes between the synchronous R1 devices and their paired R2 devices. Normal SRDF mirroring resumes. Consistency protection is automatically activated upon resumption of the link:

```
symrdf -cg srdftest2 resume -rdfg name:rdfg4 -noprompt

An RDF 'Resume' operation execution is
in progress for composite group 'srdftest2'. Please wait...
```

```
    Resume RDF link(s) for device(s) in (0150,04)..................Started.
    Resume RDF link(s) for device(s) in (0150,04)..................Done.

The RDF 'Resume' operation successfully executed for
composite group 'srdftest2'.
```

- The `symrdf verify` command displays a message every 30 seconds until all SRDF pairs in the `rdfg4` subset are synchronized:

```
symrdf -cg srdftest2 verify -rdfg name:rdfg4 -i 30 -synchronized

Not all devices in the RDF group 'srdftest2' are in the 'Synchronized' state.

Not all devices in the RDF group 'srdftest2' are in the 'Synchronized' state.

All devices in the group 'srdftest2' are in 'Synchronized' state.
```

- The `symrdf query` command confirms that the synchronous pairs in `rdfg4` have returned to the Synchronized state:

```
symrdf -cg srdftest2 query

Composite Group Name      :  srdftest2
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300180    (Microcode Version: 5771)
RDF (RA) Group Number             :  4 (03) - rdfg4
Star Mode                         : NO

Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------ ----- ------ --------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv      n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDA   s  p   STATE
-------------------------------- -- ----------------------- ----- ------ --------
DEV002   0082 RW       0       0 RW 0082 WD       0       0 S..   X  -   Synchronized
DEV001   0072 RW       0       0 RW 0072 WD       0       0 S..   X  -   Synchronized


Symmetrix ID                      : 000190300150    (Microcode Version: 5771)
Remote Symmetrix ID               : 000190300152    (Microcode Version: 5771)
RDF (RA) Group Number             :  5 (04) - rdfg5
Star Mode                         : NO

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------ ----- ------ --------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv      n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDA   s  p   STATE
-------------------------------- -- ----------------------- ----- ------ --------
DEV002   0082 RW       0       0 RW 0082 WD       0       0 A..   X  -   Consistent
DEV001   0072 RW       0       0 RW 0072 WD       0       0 A..   X  -   Consistent

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:
```

```
 M(ode of Operation): A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)           : X = Enabled, . = Disabled
 A(daptive Copy)    : D = Disk Mode, W = WP Mode, . = ACp off

Legend for STATES:

 Cons(istency State): X = Enabled, M = Mixed, . = Disabled, - = N/A
 Susp(end State)    : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Example 6: Dynamic modification of SRDF consistency groups

This example shows you how to dynamically add and remove devices from RDF1 consistency groups while still maintaining consistency protection for the following device types:

- Simple R1
- Concurrent R11
- Cascaded R1

For a comprehensive overview of this feature, see "Dynamic modification of SRDF consistency groups" on page 213.

RDF1 consistency group: Adding and removing devices

This R1->R2 example shows how to add and remove device pairs from the `simple` CG using the `drdf_stg file`, which contains the following device pairs:

```
02C0 02C0
02C1 02C1
02C2 02C2
02C3 02C3
```

R1 is on Symmetrix 88 and R2 is on Symmetrix 76.

- The following `query` shows the `simple` CG before device pairs were added to it:

```
symrdf -cg simple query

Composite Group Name      :  simple
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  -------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
--------------------------------- -- ------------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
```

```
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following dynamic modify add operation moves the file's device pairs from RDFG 3 in the staging area to RDFG 1 in the `simple` CG:

```
symcg -cg simple modify -add -sid 88 -stg_rdfg 3 -f drdf_stg -cg_rdfg 1 -nop

A consistency 'Modify_Add' operation execution is
in progress for composite group 'simple'. Please wait...

        CG modify add for selected device(s) in (3088, 001)..............Started.
        Move RDF Pair from (3088,003) to (3088,001).....................Started.
        Move RDF Pair from (3088,003) to (3088,001).....................Done.
        Resume RDF link(s) for device(s) in (3088,001)..................Started.
        Resume RDF link(s) for device(s) in (3088,001)..................Done.
        CG modify add for selected device(s) in (3088, 001)..............Done.

The consistency 'Modify_Add' operation successfully executed for
composite group 'simple'.
```

- The following `query` highlights the added device pairs in the `simple` CG after the dynamic modify add operation:

```
symrdf -cg simple query

Composite Group Name      :  simple
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------- ----- ------ ------------
```

```
               ST                 LI      ST                       C  S
  Standard      A                  N       A                       o  u
Logical   Sym    T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device    Dev    E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001    02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
DEV002    02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003    02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
DEV004    02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005    02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006    02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007    02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008    02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009    02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010    02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011    02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012    02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013    02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014    02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
DEV015    02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016    02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized
DEV017    02C0 RW       0       0 RW 02C0 WD       0       0 S...  X  -   Synchronized
DEV018    02C1 RW       0       0 RW 02C1 WD       0       0 S...  X  -   Synchronized
DEV019    02C2 RW       0       0 RW 02C2 WD       0       0 S...  X  -   Synchronized
DEV020    02C3 RW       0       0 RW 02C3 WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following dynamic modify remove operation moves the device pairs from the `simple` CG back to RDFG 3 in the staging area:

```
symcg -cg simple modify -remove -sid 88 -stg_rdfg 3 -f drdf_stg -cg_rdfg 1 -nop

A consistency 'Modify_Remove' operation execution is
in progress for composite group 'simple'. Please wait...

        CG modify remove for selected device(s) in (3088, 001)..............Started.
        Suspend RDF link(s) for device(s) in (3088,001)..................Done.
        Move RDF Pair from (3088,001) to (3088,003).....................Started.
        Move RDF Pair from (3088,003) to (3088,003).....................Done.
        CG modify remove for selected device(s) in (3088, 001)..............Done.

The consistency 'Modify_Remove' operation successfully executed for
composite group 'simple'.
```

- The following `query` shows the four device pairs are no longer in the `simple` CG:

```
symrdf -cg simple query

Composite Group Name      :  simple
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
```

```
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Cascaded RDF1 consistency group: Adding and removing devices

In this example, the cascaded configuration is as follows:

SRDF consistency group:

R1 is on Symmetrix 88. R1->R21 uses RDFG 1.
R21 is on Symmetrix 76. R21->R2 uses RDFG 2.
R2 is on Symmetrix 156.

RDF-ECA consistency protection is enabled for the R1->R21 hop through RDFG 1. The devices in the `casc` CG range from 2B0-2BF.

Staging area:

R1 is on Symmetrix 88. R1->R21 uses RDFG 3.

R21 is on Symmetrix 76. R21->R2 uses RDFG 4.
R2 is on Symmetrix 156.

Device pairs 02C0 through 02C3 will be added to the `casc` CG.

- The following dynamic modify add operation moves the file's device pairs from RDFG 3 and RDFG 4 in the staging area to RDFG 1 and RDFG 2 in the `casc` CG:

```
symcg -cg casc modify -add -sid 088 -stg_rdfg 3 -f drdf_stg -stg_r21_rdfg 4 -cg_rdfg 1
-cg_r21_rdfg 2 -nop

A consistency 'Modify_Add' operation execution is
in progress for composite group 'casc'. Please wait...

        CG modify add for selected device(s) in (3088, 001)..............Started.
        CG modify add for selected device(s) in (3076, 002)..............Started.
        Move RDF Pair from (3088,003) to (3088,001)......................Started.
        Move RDF Pair from (3088,003) to (3088,001)......................Done.
        Move RDF Pair from (3076,004) to (3076,002)......................Started.
        Move RDF Pair from (3076,004) to (3076,002)......................Done.
        Resume RDF link(s) for device(s) in (3088,001)...................Started.
        Resume RDF link(s) for device(s) in (3088,001)...................Done.
        CG modify add for selected device(s) in (3076, 002)..............Done.
        CG modify add for selected device(s) in (3088, 001)..............Done.

The consistency 'Modify_Add' operation successfully executed for
composite group 'casc'.
```

- The following `query` highlights the device pairs added to the casc CG spanning the first hop (R1->R21):

```
symrdf -cg casc query

Composite Group Name      :  casc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
--------------------------------  ------------------------ ----- ------ ------------
               ST                 LI      ST                       C  S
  Standard      A                  N       A                       o  u
Logical  Sym    T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev    E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ------------------------ ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
```

```
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized
DEV017   02C0 RW       0       0 RW 02C0 WD       0       0 S...  X  -   Synchronized
DEV018   02C1 RW       0       0 RW 02C1 WD       0       0 S...  X  -   Synchronized
DEV019   02C2 RW       0       0 RW 02C2 WD       0       0 S...  X  -   Synchronized
DEV020   02C3 RW       0       0 RW 02C3 WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following `query` highlights the devices added to the casc CG spanning the second hop (R21->R2):

```
symrdf -cg casc query -hop2

Composite Group Name      :  casc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Hop-2 Symmetrix ID                : 000192603076    (Microcode Version: 5875)
Hop-2 Remote Symmetrix ID         : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)
Hop-2 RDF (RA) Group Number       :   2 (01)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------ ----- ------ ------------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ------------------------ ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 C.D.  .  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 C.D.  .  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 C.D.  .  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 C.D.  .  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 C.D.  .  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 C.D.  .  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 C.D.  .  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 C.D.  .  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 C.D.  .  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 C.D.  .  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 C.D.  .  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 C.D.  .  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 C.D.  .  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 C.D.  .  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 C.D.  .  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 C.D.  .  -   Synchronized
DEV017   02C0 RW       0       0 NR 02C0 WD       0       0 C.D.  .  -   Suspended
DEV018   02C1 RW       0       0 NR 02C1 WD       0       0 C.D.  .  -   Suspended
```

```
DEV019   02C2 RW       0       0 NR 02C2 WD       0       0 C.D.  .  -   Suspended
DEV020   02C3 RW       0       0 NR 02C3 WD       0       0 C.D.  .  -   Suspended

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following dynamic modify remove operation moves the device pairs from the `casc` CG back to RDFG 3 and RDFG 4 in the staging area:

```
symcg -cg casc modify -remove -sid 088 -stg_rdfg 3 -f drdf_stg -stg_r21_rdfg 4 -cg_rdfg 1
-cg_r21_rdfg 2 -nop

A consistency 'Modify_Remove' operation execution is
in progress for composite group 'casc'. Please wait...

        CG modify remove for selected device(s) in (3088, 001)..............Started.
        CG modify remove for selected device(s) in (3076, 002)..............Started.
        Suspend RDF link(s) for device(s) in (3088,001)..................Done.
        Suspend RDF link(s) for device(s) in (3076,002)..................Done.
        Move RDF Pair from (3088,001) to (3088,003)......................Started.
        Move RDF Pair from (3088,003) to (3088,003)......................Done.
        Move RDF Pair from (3076,002) to (3076,004)......................Started.
        Move RDF Pair from (3076,002) to (3076,004)......................Done.
        CG modify remove for selected device(s) in (3088, 001)..............Done.
        CG modify remove for selected device(s) in (3076, 002)..............Done.

The consistency 'Modify_Remove' operation successfully executed for
composite group 'casc'.
```

- The following `query` shows the casc CG spanning hop 1 after the dynamic modify remove operation:

```
symrdf -cg casc query

Composite Group Name      :  casc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  SYNC


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
------------------------------- -- ----------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
```

```
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

- The following `query` shows the casc CG spanning hop 2 after the dynamic modify remove operation:

symrdf -cg casc query -hop2

```
Composite Group Name      :  casc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  1
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Hop-2 Symmetrix ID                : 000192603076    (Microcode Version: 5875)
Hop-2 Remote Symmetrix ID         : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)
Hop-2 RDF (RA) Group Number       :   2 (01)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                      C  S
  Standard     A                  N       A                      o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
-------------------------------- -- ----------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 C.D.  .  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 C.D.  .  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 C.D.  .  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 C.D.  .  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 C.D.  .  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 C.D.  .  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 C.D.  .  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 C.D.  .  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 C.D.  .  -   Synchronized
```

```
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 C.D.  .  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 C.D.  .  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 C.D.  .  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 C.D.  .  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 C.D.  .  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 C.D.  .  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 C.D.  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Concurrent RDF1 consistency group: Adding and removing devices

In this example, the concurrent configuration is as follows:

SRDF consistency group:

R11 is on Symmetrix 88.
The first R2 target is on Symmetrix 76. This R11->R2 target uses RDFG 1.
The second R2 target is on Symmetrix 156. This R11->R2 target uses RDFG 2.

Staging area:

R11 is on Symmetrix 88.
The first R2 target is on Symmetrix 76. R11->R2 uses RDFG 3.
The second R2 target is on Symmetrix 156. R11->R2 uses RDFG 4.

Device pairs 02C0 through 02C3 will be added into the `conc` CG.

- The following `query` shows that `leg1` of the `conc` CG is enabled for RDF-ECA consistency protection:

```
symrdf -cg conc query -detail

Composite Group Name      :  conc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names:
    {
    RDFG Name                          : leg1
    RDF Consistency Mode               : SYNC
    Sync Consistency Info
        {
        Consistency State              : Synchronized
        }

    RDFG Name                          : leg2
```

```
    RDF Consistency Mode                : NONE

Symmetrix ID                        : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID                 : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number               :   1 (00) - leg1

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S..X. Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S..X. Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S..X. Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S..X. Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S..X. Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S..X. Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S..X. Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S..X. Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S..X. Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S..X. Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S..X. Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S..X. Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S..X. Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S..X. Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S..X. Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S..X. Synchronized

Symmetrix ID                        : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID                 : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number               :   2 (01) - leg2

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S.... Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S.... Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S.... Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S.... Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S.... Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S.... Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S.... Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S.... Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S.... Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S.... Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S.... Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S.... Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S.... Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S.... Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S.... Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S.... Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
```

```
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following dynamic modify add operation moves the file's device pairs from RDFG 3 and RDFG 4 in the staging area to RDFG 1 and RDFG 2 in the `conc` CG:

```
symcg -cg conc modify -add -sid 88 -stg_rdfg 3,4 -f drdf_stg -cg_rdfg 1,2 -nop

A consistency 'Modify_Add' operation execution is
in progress for composite group 'conc'. Please wait...

        CG modify add for selected device(s) in (3088, 001)..............Started.
        CG modify add for selected device(s) in (3088, 002)..............Started.
        Suspend RDF link(s) for device(s) in (3088,003)..................Done.
        Suspend RDF link(s) for device(s) in (3088,004)..................Done.
        Move RDF Pair from (3088,003) to (3088,001)......................Started.
        Move RDF Pair from (3088,003) to (3088,001)......................Done.
        Move RDF Pair from (3088,004) to (3088,002)......................Started.
        Move RDF Pair from (3088,004) to (3088,002)......................Done.
        Resume RDF link(s) for device(s) in (3088,001)...................Started.
        Resume RDF link(s) for device(s) in (3088,001)...................Done.
        CG modify add for selected device(s) in (3088, 001)..............Done.
        CG modify add for selected device(s) in (3088, 002)..............Done.

The consistency 'Modify_Add' operation successfully executed for
composite group 'conc'.
```

- The following `query` highlights the devices added to the `conc` CG:

```
symrdf -cg conc query -detail

Composite Group Name      :  conc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE

RDFG Names:
    {
    RDFG Name                          : leg1
    RDF Consistency Mode               : SYNC
    Sync Consistency Info
        {
        Consistency State              : Synchronized
        }

    RDFG Name                          : leg2
    RDF Consistency Mode               : NONE
    }

Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00) - leg1

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ----------------------- ----- ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S..X. Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S..X. Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S..X. Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S..X. Synchronized
```

```
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S..X. Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S..X. Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S..X. Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S..X. Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S..X. Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S..X. Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S..X. Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S..X. Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S..X. Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S..X. Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S..X. Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S..X. Synchronized
DEV017   02C0 RW       0       0 RW 02C0 WD       0       0 S..X. Synchronized
DEV018   02C1 RW       0       0 RW 02C1 WD       0       0 S..X. Synchronized
DEV019   02C2 RW       0       0 RW 02C2 WD       0       0 S..X. Synchronized
DEV020   02C3 RW       0       0 RW 02C3 WD       0       0 S..X. Synchronized

Symmetrix ID                     : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID              : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number            :   2 (01) - leg2

        Source (R1) View             Target (R2) View       MODES
--------------------------------  ------------------------- ----- ------------
              ST                 LI      ST
  Standard     A                  N       A
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDACE STATE
-------------------------------- -- ------------------------- ----- ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S.... Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S.... Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S.... Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S.... Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S.... Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S.... Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S.... Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S.... Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S.... Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S.... Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S.... Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S.... Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S.... Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S.... Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S.... Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S.... Synchronized
DEV017   02C0 RW       0       0 NR 02C0 WD       0       0 C.D.. Suspended
DEV018   02C1 RW       0       0 NR 02C1 WD       0       0 C.D.. Suspended
DEV019   02C2 RW       0       0 NR 02C2 WD       0       0 C.D.. Suspended
DEV020   02C3 RW       0       0 NR 02C3 WD       0       0 C.D.. Suspended

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 C(onsistency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A
```

- The following dynamic modify remove operation moves the device pairs from the `conc` CG back to RDFG 3 and RDFG 4 in the staging area:

```
symcg -cg conc modify -remove -sid 88 -stg_rdfg 3,4 -f drdf_stg -cg_rdfg 1,2 -nop
```

```
A consistency 'Modify_Remove' operation execution is
in progress for composite group 'conc'. Please wait...

        CG modify remove for selected device(s) in (3088, 001)..............Started.
        CG modify remove for selected device(s) in (3088, 002)..............Started.
        Suspend RDF link(s) for device(s) in (3088,001)..................Done.
        Suspend RDF link(s) for device(s) in (3088,002)..................Done.
        Move RDF Pair from (3088,001) to (3088,003)......................Started.
        Move RDF Pair from (3088,003) to (3088,003)......................Done.
        Move RDF Pair from (3088,002) to (3088,004)......................Started.
        Move RDF Pair from (3088,002) to (3088,004)......................Done.
        CG modify remove for selected device(s) in (3088, 001)..............Done.
        CG modify remove for selected device(s) in (3088, 002)..............Done.
```

- The following `query` shows the `conc` CG after the dynamic modify remove operation:

symrdf -cg conc query

```
Composite Group Name      :  conc
Composite Group Type      :  RDF1
Number of Symmetrix Units :  1
Number of RDF (RA) Groups :  2
RDF Consistency Mode      :  NONE


Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603076    (Microcode Version: 5875)
RDF (RA) Group Number             :   1 (00)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
------------------------------- -- ----------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  X  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  X  -   Synchronized
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  X  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  X  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  X  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  X  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  X  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  X  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  X  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  X  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  X  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  X  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  X  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  X  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  X  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  X  -   Synchronized

Symmetrix ID                      : 000192603088    (Microcode Version: 5875)
Remote Symmetrix ID               : 000192603156    (Microcode Version: 5875)
RDF (RA) Group Number             :   2 (01)

        Source (R1) View             Target (R2) View       MODES STATES
-------------------------------  ------------------------- ----- ------ ------------
              ST                 LI      ST                       C  S
  Standard     A                  N       A                       o  u
Logical  Sym   T  R1 Inv  R2 Inv  K       T  R1 Inv  R2 Inv       n  s   RDF Pair
Device   Dev   E  Tracks Tracks   S Dev   E  Tracks  Tracks MDAE  s  p   STATE
------------------------------- -- ----------------------- ----- ------ ------------
DEV001   02B0 RW       0       0 RW 02B0 WD       0       0 S...  .  -   Synchronized
DEV002   02B1 RW       0       0 RW 02B1 WD       0       0 S...  .  -   Synchronized
```

```
DEV003   02B2 RW       0       0 RW 02B2 WD       0       0 S...  .  -   Synchronized
DEV004   02B3 RW       0       0 RW 02B3 WD       0       0 S...  .  -   Synchronized
DEV005   02B4 RW       0       0 RW 02B4 WD       0       0 S...  .  -   Synchronized
DEV006   02B5 RW       0       0 RW 02B5 WD       0       0 S...  .  -   Synchronized
DEV007   02B6 RW       0       0 RW 02B6 WD       0       0 S...  .  -   Synchronized
DEV008   02B7 RW       0       0 RW 02B7 WD       0       0 S...  .  -   Synchronized
DEV009   02B8 RW       0       0 RW 02B8 WD       0       0 S...  .  -   Synchronized
DEV010   02B9 RW       0       0 RW 02B9 WD       0       0 S...  .  -   Synchronized
DEV011   02BA RW       0       0 RW 02BA WD       0       0 S...  .  -   Synchronized
DEV012   02BB RW       0       0 RW 02BB WD       0       0 S...  .  -   Synchronized
DEV013   02BC RW       0       0 RW 02BC WD       0       0 S...  .  -   Synchronized
DEV014   02BD RW       0       0 RW 02BD WD       0       0 S...  .  -   Synchronized
DEV015   02BE RW       0       0 RW 02BE WD       0       0 S...  .  -   Synchronized
DEV016   02BF RW       0       0 RW 02BF WD       0       0 S...  .  -   Synchronized

Total            ------- -------            ------- -------
  Track(s)             0       0                  0       0
  MBs                0.0     0.0                0.0     0.0

Legend for MODES:

 M(ode of Operation)   : A = Async, S = Sync, E = Semi-sync, C = Adaptive Copy
 D(omino)              : X = Enabled, . = Disabled
 A(daptive Copy)       : D = Disk Mode, W = WP Mode, . = ACp off
 (Consistency) E(xempt): X = Enabled, . = Disabled, M = Mixed, - = N/A

Legend for STATES:

 Cons(istency State)   : X = Enabled, . = Disabled, M = Mixed, - = N/A
 Susp(end State)       : X = Online, . = Offline, P = Offline Pending, - = N/A
```

Example 7: Recovering from a failed dynamic add operation

In this example, the local host had a trip event while the dynamic modify add operation was running. This event caused the command to terminate, requiring a recovery operation. A recovery either completes the modify add operation or rolls back all actions performed by modify add before the failure, placing the CG into its original state. This example illustrates both types of recovery for a dynamic modify operation.

- The following command attempts to dynamically add devices `A3` and `A7` to the `metacg` CG but fails because of a trip event, requiring a recovery on this CG:

```
symcg -cg metacg -dev A3,A7 -stg_rdfg 110 -cg_rdfg 100 modify -add -nop -sid 44

A consistency 'Modify_Add' operation execution is
in progress for composite group 'metacg'. Please wait...

        CG modify add for selected device(s) in (0144, 100).............Started
        Suspend RDF link(s) for device(s) in (0144,110).................Not Needed.
        Move RDF Pair from (0144,110) to (0144,100).....................Started.
        Move RDF Pair from (0144,110) to (0144,100).....................Started.
        Move RDF Pair from (0144,110) to (0144,100).....................Started.
        Move RDF Pair from (0144,110) to (0144,100).....................Started.

The dynamic RDF operation failed,see the SYMAPI log file for more information
```

- The following establish operation cannot complete because the `metacg` CG requires a recovery:

```
symrdf -cg metacg est -v -nop

An RDF 'Incremental Establish' operation execution is
in progress for composite group 'metacg'. Please wait...
```

The CG Modify recovery action is required.

- The following `symcg modify -recover` command successfully completed the modify add operation:

```
symcg -cg metacg  modify -recover -nop

A consistency 'Modify_Recover' operation execution is
in progress for composite group 'metacg'. Please wait...

        CG modify add for selected device(s) in (0144, 100).............Started.
        Suspend RDF link(s) for device(s) in (0144,110)................Not Needed.
        Move RDF Pair from (0144,110) to (0144,100).....................Started.
        Move RDF Pair from (0144,110) to (0144,100).....................Done.
        Set Mode Sync for device(s) in (0144,100).......................Started.
        Set Mode Sync for device(s) in (0144,100).......................Done.
        Add selected device(s) in (0144,100) to CG......................Started.
        Add selected device(s) in (0144,100) to CG......................Done.
        Enable RDF Consistency for selected device(s) in (0144,100).....Started.
        Enable RDF Consistency for selected device(s) in (0144,100).....Done.
        CG modify add for selected device(s) in (0144, 100).............Done.

The consistency 'Modify_Recover' operation successfully executed for composite group 'metacg'.
```

- As shown in the following output, devices A3 and A7 are now part of RDFG 100, and belong to `metacg`:

```
symcg show metacg

Composite Group Name:  metacg

  Composite Group Type                                : RDF1
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : No
  RDF Consistency Protection Allowed                  : Yes
  RDF Consistency Mode                                : SYNC
  Concurrent RDF                                      : No
  Cascaded RDF                                        : No

  Number of RDF (RA) Groups                           :    1
  Number of STD Devices                               :    6
  Number of CRDF STD Devices                          :    0
  Number of BCV's (Locally-associated)                :    0
  Number of VDEV's (Locally-associated)               :    0
  Number of TGT's Locally-associated                  :    0
  Number of CRDF TGT Devices                          :    0
  Number of RVDEV's (Remotely-associated VDEV)        :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)     :    0
  Number of RRBCV's (Remotely-associated RBCV)        :    0
  Number of RTGT's (Remotely-associated)              :    0
  Number of Hop2 BCV's (Remotely-assoc'ed Hop2 BCV)   :    0
  Number of Hop2 VDEV's (Remotely-assoc'ed Hop2 VDEV):    0
  Number of Hop2 TGT's (Remotely-assoc'ed Hop2 TGT)   :    0
  Number of Device Groups                             :    0
  Device Group Names                                  : N/A

  Number of Symmetrix Units (1):
    {

    1) Symmetrix ID                                    : 000194900144
       Microcode Version                               : 5875
       Number of STD Devices                           :    6
       Number of CRDF STD Devices                      :    0
```

```
      Number of BCV's (Locally-associated)           :    0
      Number of VDEV's (Locally-associated)          :    0
      Number of TGT's Locally-associated             :    0
      Number of CRDF TGT Devices                     :    0
      Number of RVDEV's (Remotely-associated VDEV)   :    0
      Number of RBCV's (Remotely-associated STD_RDF) :    0
      Number of BRBCV's (Remotely-associated BCV-RDF):    0
      Number of RTGT's (Remotely-associated)         :    0
      Number of RRBCV's (Remotely-associated RBCV)   :    0
      Number of Hop2BCV's (Remotely-assoc'ed Hop2BCV):    0
      Number of Hop2VDEVs(Remotely-assoc'ed Hop2VDEV):    0
      Number of Hop2TGT's (Remotely-assoc'ed Hop2TGT):    0

      Number of RDF (RA) Groups (1):
        {

        1) RDF (RA) Group Number  : 100            (63)
           Remote Symmetrix ID    : 000192600237
           Microcode Version      : 5875
           Recovery RA Group      : N/A            (N/A)
           RA Group Name          : N/A

           STD Devices (6):
             {
             ------------------------------------------------------------------------------
                                                Sym  Device             Flags     Cap
             LdevName    PdevName               Dev  Config        Sts   CSRT     (MB)
             ------------------------------------------------------------------------------
             DEV001      /dev/sdch              0083 RDF1+Mir      RW    X--1    34523
             DEV002      /dev/sdci              0087 RDF1+Mir      RW    X--1    34523
             DEV003      /dev/sdcj              008B RDF1+Mir      RW    X--1    34523
             DEV004      /dev/sdck              008C RDF1+Mir      RW    X--1    34523
             DEV005      N/A                    00A3 RDF1+Mir      RW    X--1    34523
             DEV006      N/A                    00A7 RDF1+Mir      RW    X--1    34523
Legend:
  RDFA Flags:
      C(onsistency)   : X = Enabled, . = Disabled, - = N/A
      (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
      R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
      (Mirror) T(ype) : 1 = R1, 2 = R2, - = N/A
```

- The following `symcg -recover` command with the `-force` option was unable to complete the symcg modify add operation, and all changes made to `metacg` were rolled back. The highlighted sentence informs you that a recover rollback was performed on the modify add operation, placing `metacg` into its original state.

```
symcg -cg metacg  modify -recover -nop -force

A consistency 'Modify_Recover' operation execution is
in progress for composite group 'metacg'. Please wait...

        CG modify add for selected device(s) in (0144, 100).............Started.
        Suspend RDF link(s) for device(s) in (0144,110)................Not Needed.
        Half Move RDF Pair from (0237,110) to (0237,100)...............Not Needed.
        Half Move RDF Pair from (0144,110) to (0144,100)...............Not Needed.
      .
      .
      .
   CG modify add (rollback) for selected device(s) in (0144, 100)...Done.

The consistency 'Modify_Recover' operation successfully executed for composite group 'metacg'.
```

- The following shows that `metacg` retained its original state. Neither device A3 nor A7 were added to RDFG 100:

symcg show metacg

```
Composite Group Name:  metacg

  Composite Group Type                                : RDF1
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : No
  RDF Consistency Protection Allowed                  : Yes
  RDF Consistency Mode                                : SYNC
  Concurrent RDF                                      : No
  Cascaded RDF                                        : No

  Number of RDF (RA) Groups                           :    1
  Number of STD Devices                               :    4
  Number of CRDF STD Devices                          :    0
  Number of BCV's (Locally-associated)                :    0
  Number of VDEV's (Locally-associated)               :    0
  Number of TGT's Locally-associated                  :    0
  Number of CRDF TGT Devices                          :    0
  Number of RVDEV's (Remotely-associated VDEV)        :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)     :    0
  Number of RRBCV's (Remotely-associated RBCV)        :    0
  Number of RTGT's (Remotely-associated)              :    0
  Number of Hop2 BCV's (Remotely-assoc'ed Hop2 BCV)   :    0
  Number of Hop2 VDEV's (Remotely-assoc'ed Hop2 VDEV):    0
  Number of Hop2 TGT's (Remotely-assoc'ed Hop2 TGT)   :    0
  Number of Device Groups                             :    0
  Device Group Names                                  : N/A

  Number of Symmetrix Units (1):
    {

    1) Symmetrix ID                                    : 000194900144
       Microcode Version                               : 5875
       Number of STD Devices                           :    4
       Number of CRDF STD Devices                      :    0
       Number of BCV's (Locally-associated)            :    0
       Number of VDEV's (Locally-associated)           :    0
       Number of TGT's Locally-associated              :    0
       Number of CRDF TGT Devices                      :    0
       Number of RVDEV's (Remotely-associated VDEV)    :    0
       Number of RBCV's (Remotely-associated STD_RDF)  :    0
       Number of BRBCV's (Remotely-associated BCV-RDF):    0
       Number of RTGT's (Remotely-associated)          :    0
       Number of RRBCV's (Remotely-associated RBCV)    :    0
       Number of Hop2BCV's (Remotely-assoc'ed Hop2BCV):    0
       Number of Hop2VDEVs(Remotely-assoc'ed Hop2VDEV):    0
       Number of Hop2TGT's (Remotely-assoc'ed Hop2TGT):    0

       Number of RDF (RA) Groups (1):
         {

         1) RDF (RA) Group Number  : 100            (63)
            Remote Symmetrix ID    : N/A
            Microcode Version      : N/A
            Recovery RA Group      : N/A            (N/A)
            RA Group Name          : N/A

            STD Devices (4):
              {
              ------------------------------------------------------------------------------
                                                  Sym  Device             Flags     Cap
              LdevName    PdevName                Dev  Config        Sts   CSRT     (MB)
              ------------------------------------------------------------------------------
```

```
              DEV001      /dev/sdch                 0083 RDF1+Mir      RW    X--1    34523
              DEV002      /dev/sdci                 0087 RDF1+Mir      RW    X--1    34523
              DEV003      /dev/sdcj                 008B RDF1+Mir      RW    X--1    34523
              DEV004      /dev/sdck                 008C RDF1+Mir      RW    X--1    34523

Legend:
  RDFA Flags:
       C(onsistency)   : X = Enabled, . = Disabled, - = N/A
       (RDFA) S(tatus) : A = Active, I = Inactive, - = N/A
       R(DFA Mode)     : S = Single-session mode, M = MSC mode, - = N/A
       (Mirror) T(ype) : 1 = R1, 2 = R2, - = N/A
```

CHAPTER 16
Performing SRDF/Automated Replication Operations

This chapter provides SYMCLI examples of specific actions and commands, which replicate data in pre-defined cycles using the SRDF automated replication process.

Examples include:

- Example 1: SRDF/AR single-hop configuration 424
- Example 2: SRDF/AR multi-hop configuration with BCVs at hop 2 433
- Example 3: SRDF/AR single-hop configuration using a CG 435
- Example 4: SRDF/AR multi-hop configuration using a CG 442
- Example 5: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (single-hop configuration) 447
- Example 6: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (multi-hop configuration) 451
- Example 7: Restarting a replicate session when devices are locked 456

Note: Some of the examples in this section were executed with lower versions of software. Therefore, your output displays may not look exactly like the ones appearing in these examples.

Example 1: SRDF/AR single-hop configuration

This example is performed using:

- Solutions Enabler version 5.4.
- The hardware setup consists of a HP-UX host connected to a source Symmetrix (sid 161).
- All commands are issued from the source-side host.

The example uses the following Symmetrix systems to create the single-hop environment:

- Local Source Symmetrix (sid 35) with:
  - Standard devices 56-6E
  - R1 BCV devices 182-19A
- Remote Target Symmetrix (sid 41): BCV devices 137-14F

The following `symcfg list` command displays the Symmetrix arrays attached to this host. Symmetrix arrays 000187900035 and 000187900041 are configured as a single-hop configuration:

```
symcfg list

                        S Y M M E T R I X

                                    Mcode    Cache      Num Phys  Num
Symm
 SymmID       Attachment  Model     Version  Size (MB)  Devices
Devices

 000000003143 Local       3630      5267       4096     209       447
 000000005232 Local       8230      5568       2048       6       652
 000184500160 Local       8430      5568      12288     132      2054
 000187700079 Local       DMX2000P  5670      51200     189      3326
 000187900035 Local       DMX800    5670       6144     204       812
 000000003156 Remote      3630      5267       4096       0       448
 000000005231 Remote      8230      5568       4096       0       714
 000000005233 Remote      8230      5568       6144       0       122
 000000006201 Remote      DMX2000P  5670      16384       0       824
 000184502898 Remote      8530      5568      12288       0      1139
 000187700067 Remote      2000P-M2  5670      51200       0      3762
 000187900041 Remote      DMX800    5670       6144       0       721
```

- The `symdg create` command creates a device group named `symrep`:

```
symdg create symrep
```

- The `sympd list` command displays all Symmetrix devices that are visible to this host.

  The display below has been edited to show those devices that are used in the example. The **N/Grp'd** attribute means that these devices are not already part of a device group and are free to be added to device group `symrep`:

```
sympd list -sid 35

Symmetrix ID: 000187900035

        Device Name           Directors                   Device
--------------------------- ------------- ----------------------------------
                                                                        Cap
Physical                Sym  SA :P DA :IT  Config        Attribute    Sts   (MB)
```

```
--------------------------- ------------- ----------------------------------

/dev/rdsk/c15t2d0s2    0008 01C:1 01A:CC  2-Way Mir     N/Grp'd      RW 3
/dev/rdsk/c15t2d1s2    0009 01C:1 16B:CC  2-Way Mir     N/Grp'd      RW 3
/dev/rdsk/c15t2d2s2    0056 01C:1 01B:CD  2-Way Mir     N/Grp'd  (M) RW 12946
/dev/rdsk/c15t2d3s2    0059 01C:1 16A:C5  2-Way Mir     N/Grp'd  (M) RW 8631
/dev/rdsk/c15t2d4s2    005B 01C:1 16B:C2  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d5s2    005C 01C:1 01A:CA  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d6s2    005D 01C:1 16A:C7  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d7s2    005E 01C:1 01B:C7  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d8s2    005F 01C:1 16B:C4  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d9s2    0060 01C:1 01A:C6  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d10s2   0061 01C:1 16A:C9  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d11s2   0062 01C:1 01B:C9  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d12s2   0063 01C:1 16B:C6  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d13s2   0064 01C:1 01A:C8  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d14s2   0065 01C:1 02A:CB  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d15s2   0066 01C:1 15B:CB  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d16s2   0067 01C:1 02B:CA  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d17s2   0068 01C:1 15A:CC  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d18s2   0069 01C:1 02A:CD  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d19s2   006A 01C:1 15B:C1  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d20s2   006B 01C:1 02B:C0  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d21s2   006C 01C:1 15A:CE  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d22s2   006D 01C:1 02A:C1  2-Way Mir     N/Grp'd      RW 4315
/dev/rdsk/c15t2d23s2   006E 01C:1 15B:C3  2-Way Mir     N/Grp'd      RW 4315
```

- The `symdg` command adds one or more standard devices to the device group.

  The `-range` option can be used with the `addall` action to limit the selection to the devices that are within the specified range (for example, devices 56 through 6E):

  `symdg -g symrep addall dev -range 56:6E -sid 35`

- The `symdev list` command with the `-r1 -bcv` options displays those R1 BCV devices in the local Symmetrix system (sid 35) that are not already part of a device group (**N/Asst'd**) and which are free to be added to the device group.

  The display below has been edited to show those devices that are used in the example:

`symdev list -r1 -bcv -sid 35`

```
Symmetrix ID: 000187900035

        Device Name           Directors                  Device
--------------------------- ------------- ----------------------------------
                                                                        Cap
Sym  Physical               SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- ----------------------------------

0182 Not Visible            ???:? 16A:CB  RDF1-BCV+Mir  N/Asst'd (M) NR 12946
0183 Not Visible            ???:? 01B:CB  RDF1-BCV+Mir  N/Asst'd (m) NR       -
0184 Not Visible            ???:? 16B:CA  RDF1-BCV+Mir  N/Asst'd (m) NR       -
0185 Not Visible            ???:? 01A:CC  RDF1-BCV+Mir  N/Asst'd (M) NR 8631
0186 Not Visible            ???:? 16A:CD  RDF1-BCV+Mir  N/Asst'd (m) NR       -
0187 Not Visible            ???:? 01B:C1  RDF1-BCV+Mir  N/Asst'd     NR    4315
0188 Not Visible            ???:? 16B:C0  RDF1-BCV+Mir  N/Asst'd     NR    4315
0189 Not Visible            ???:? 01A:CE  RDF1-BCV+Mir  N/Asst'd     NR    4315
018A Not Visible            ???:? 16A:C1  RDF1-BCV+Mir  N/Asst'd     NR    4315
018B Not Visible            ???:? 01B:C3  RDF1-BCV+Mir  N/Asst'd     NR    4315
018C Not Visible            ???:? 16B:C8  RDF1-BCV+Mir  N/Asst'd     NR    4315
018D Not Visible            ???:? 01A:C0  RDF1-BCV+Mir  N/Asst'd     NR    4315
018E Not Visible            ???:? 16A:C3  RDF1-BCV+Mir  N/Asst'd     NR    4315
018F Not Visible            ???:? 01B:CD  RDF1-BCV+Mir  N/Asst'd     NR    4315
```

```
0190 Not Visible            ???:? 16B:CC  RDF1-BCV+Mir  N/Asst'd     NR    4315
0191 Not Visible            ???:? 01A:C2  RDF1-BCV+Mir  N/Asst'd     NR    4315
0192 Not Visible            ???:? 16A:C5  RDF1-BCV+Mir  N/Asst'd     NR    4315
0193 Not Visible            ???:? 01B:C5  RDF1-BCV+Mir  N/Asst'd     NR    4315
0194 Not Visible            ???:? 16B:CE  RDF1-BCV+Mir  N/Asst'd     NR    4315
0195 Not Visible            ???:? 01A:C4  RDF1-BCV+Mir  N/Asst'd     NR    4315
0196 Not Visible            ???:? 16A:C7  RDF1-BCV+Mir  N/Asst'd     NR    4315
0197 Not Visible            ???:? 01B:C7  RDF1-BCV+Mir  N/Asst'd     NR    4315
0198 Not Visible            ???:? 16B:C2  RDF1-BCV+Mir  N/Asst'd     NR    4315
0199 Not Visible            ???:? 01A:CA  RDF1-BCV+Mir  N/Asst'd     NR    4315
019A Not Visible            ???:? 16A:C9  RDF1-BCV+Mir  N/Asst'd     NR    4315
```

- The `symbcv` command associates the local BCV devices with the device group.

  The `-range` option is used with the `associate all` action to limit the selection to those BCVs that are within the specified range:

  symbcv -g symrep associateall dev -range 182:19A

- The `symdev list` command with the `-bcv` option displays those BCV devices in the remote Symmetrix (sid 41) that are not already part of a device group (**N/Asst'd**) and which are free to be added to the device group.

  The display below has been edited to show those devices that are used in the example:

symdev list -sid 41 -bcv

```
Symmetrix ID: 000187900041

        Device Name           Directors                   Device
---------------------------- ------------- ----------------------------------
                                                                        Cap
Sym  Physical                SA :P DA :IT  Config        Attribute    Sts   (MB)
---------------------------- ------------- ----------------------------------

0137 Not Visible             01C:0 02B:C8  BCV           N/Asst'd (M) RW   12946
0138 Not Visible             01C:0 01A:C8  BCV           N/Asst'd (m) RW       -
0139 Not Visible             01C:0 15A:C8  BCV           N/Asst'd (m) RW       -
013A Not Visible             01C:0 16A:CB  BCV           N/Asst'd (M) RW    8631
013B Not Visible             01C:0 02A:CB  BCV           N/Asst'd (m) RW       -
013C Not Visible             01C:0 01B:C7  BCV           N/Asst'd     RW    4315
013D Not Visible             01C:0 15B:C7  BCV           N/Asst'd     RW    4315
013E Not Visible             01C:0 16B:C0  BCV           N/Asst'd     RW    4315
013F Not Visible             01C:0 02B:C0  BCV           N/Asst'd     RW    4315
0140 Not Visible             01C:0 01A:C0  BCV           N/Asst'd     RW    4315
0141 Not Visible             01C:0 15A:C0  BCV           N/Asst'd     RW    4315
0142 Not Visible             01C:0 16A:CD  BCV           N/Asst'd     RW    4315
0143 Not Visible             01C:0 02A:CD  BCV           N/Asst'd     RW    4315
0144 Not Visible             01C:0 01B:CB  BCV           N/Asst'd     RW    4315
0145 Not Visible             01C:0 15B:CB  BCV           N/Asst'd     RW    4315
0146 Not Visible             01C:0 16B:C6  BCV           N/Asst'd     RW    4315
0147 Not Visible             01C:0 02B:C6  BCV           N/Asst'd     RW    4315
0148 Not Visible             01C:0 01A:C2  BCV           N/Asst'd     RW    4315
0149 Not Visible             01C:0 15A:C2  BCV           N/Asst'd     RW    4315
014A Not Visible             01C:0 16A:C1  BCV           N/Asst'd     RW    4315
014B Not Visible             01C:0 02A:C1  BCV           N/Asst'd     RW    4315
014C Not Visible             01C:0 01B:CD  BCV           N/Asst'd     RW    4315
014D Not Visible             01C:0 15B:CD  BCV           N/Asst'd     RW    4315
014E Not Visible             01C:0 16B:CA  BCV           N/Asst'd     RW    4315
014F Not Visible             01C:0 02B:CA  BCV           N/Asst'd     RW    4315
```

- The `symbcv` command with the `-rdf` and `-bcv` options associates the *remote* BCV devices with the device group:

```
symbcv -g symrep associateall dev -range 137:14F -bcv -rdf
```

- The `symdg show` command displays detailed group information about device group `symrep`.

  The group contains 22 local standard devices, 22 local R1 BCVs, and 22 remote BCVs:

```
symdg show symrep

Group Name:  symrep

    Group Type                                    : REGULAR
    Device Group in GNS                           : No
    Valid                                         : Yes
    Symmetrix ID                                  : 000187900035
    Group Creation Time                           : Mon Nov 17 11:27:08 2003
    Vendor ID                                     : EMC Corp
    Application ID                                : SYMCLI

    Number of STD Devices in Group                :   22
    Number of Associated GK's                     :    0
    Number of Locally-associated BCV's            :   22
    Number of Locally-associated VDEV's           :    0
    Number of Remotely-associated BCV's (STD RDF):    0
    Number of Remotely-associated BCV's (BCV RDF):   22
    Number of Remotely-assoc'd RBCV's (RBCV RDF) :    0

    Standard (STD) Devices (22):
        {
        --------------------------------------------------------------------
                                                     Sym                Cap
        LdevName              PdevName               Dev  Att. Sts      (MB)
        --------------------------------------------------------------------
        DEV001                /dev/vx/rdmp/c15t2d2s2 0056 (M)  RW      12946
        DEV002                /dev/vx/rdmp/c15t2d3s2 0059 (M)  RW       8631
        DEV003                /dev/vx/rdmp/c15t2d4s2 005B      RW       4315
        DEV004                /dev/vx/rdmp/c15t2d5s2 005C      RW       4315
        DEV005                /dev/vx/rdmp/c15t2d6s2 005D      RW       4315
        DEV006                /dev/vx/rdmp/c15t2d7s2 005E      RW       4315
        DEV007                /dev/vx/rdmp/c15t2d8s2 005F      RW       4315
        DEV008                /dev/vx/rdmp/c15t2d9s2 0060      RW       4315
        DEV009                /dev/vx/rdmp/c15t2d10s2 0061     RW       4315
        DEV010                /dev/vx/rdmp/c15t2d11s2 0062     RW       4315
        DEV011                /dev/vx/rdmp/c15t2d12s2 0063     RW       4315
        DEV012                /dev/vx/rdmp/c15t2d13s2 0064     RW       4315
        DEV013                /dev/vx/rdmp/c15t2d14s2 0065     RW       4315
        DEV014                /dev/vx/rdmp/c15t2d15s2 0066     RW       4315
        DEV015                /dev/vx/rdmp/c15t2d16s2 0067     RW       4315
        DEV016                /dev/vx/rdmp/c15t2d17s2 0068     RW       4315
        DEV017                /dev/vx/rdmp/c15t2d18s2 0069     RW       4315
        DEV018                /dev/vx/rdmp/c15t2d19s2 006A     RW       4315
        DEV019                /dev/vx/rdmp/c15t2d20s2 006B     RW       4315
        DEV020                /dev/vx/rdmp/c15t2d21s2 006C     RW       4315
        DEV021                /dev/vx/rdmp/c15t2d22s2 006D     RW       4315
        DEV022                /dev/vx/rdmp/c15t2d23s2 006E     RW       4315
        }

    BCV Devices Locally-associated (22):
        {
        --------------------------------------------------------------------
                                                     Sym                Cap
        LdevName              PdevName               Dev  Att. Sts      (MB)
        --------------------------------------------------------------------
        BCV001                N/A                    0182 (M)  NR      12946
        BCV002                N/A                    0185 (M)  NR       8631
        BCV003                N/A                    0187      NR       4315
```

```
    BCV004                N/A                     0188      NR      4315
    BCV005                N/A                     0189      NR      4315
    BCV006                N/A                     018A      NR      4315
    BCV007                N/A                     018B      NR      4315
    BCV008                N/A                     018C      NR      4315
    BCV009                N/A                     018D      NR      4315
    BCV010                N/A                     018E      NR      4315
    BCV011                N/A                     018F      NR      4315
    BCV012                N/A                     0190      NR      4315
    BCV013                N/A                     0191      NR      4315
    BCV014                N/A                     0192      NR      4315
    BCV015                N/A                     0193      NR      4315
    BCV016                N/A                     0194      NR      4315
    BCV017                N/A                     0195      NR      4315
    BCV018                N/A                     0196      NR      4315
    BCV019                N/A                     0197      NR      4315
    BCV020                N/A                     0198      NR      4315
    BCV021                N/A                     0199      NR      4315
    BCV022                N/A                     019A      NR      4315
    }

BCV Devices Remotely-associated (BCV RDF) (22):
    {
    ----------------------------------------------------------------------
                                                  Sym                  Cap
    LdevName              PdevName                Dev  Att. Sts       (MB)
    ----------------------------------------------------------------------
    BRBCV001              N/A                     0137 (M)  RW      12946
    BRBCV002              N/A                     013A (M)  RW       8631
    BRBCV003              N/A                     013C      RW       4315
    BRBCV004              N/A                     013D      RW       4315
    BRBCV005              N/A                     013E      RW       4315
    BRBCV006              N/A                     013F      RW       4315
    BRBCV007              N/A                     0140      RW       4315
    BRBCV008              N/A                     0141      RW       4315
    BRBCV009              N/A                     0142      RW       4315
    BRBCV010              N/A                     0143      RW       4315
    BRBCV011              N/A                     0144      RW       4315
    BRBCV012              N/A                     0145      RW       4315
    BRBCV013              N/A                     0146      RW       4315
    BRBCV014              N/A                     0147      RW       4315
    BRBCV015              N/A                     0148      RW       4315
    BRBCV017              N/A                     014A      RW       4315
    BRBCV018              N/A                     014B      RW       4315
    BRBCV019              N/A                     014C      RW       4315
    BRBCV020              N/A                     014D      RW       4315
    BRBCV021              N/A                     014E      RW       4315
    BRBCV022              N/A                     014F      RW       4315
    }

Device Group BCV RDF Information
    {
    RDF Type                                : R1
    RDF (RA) Group Number                   : 1               (00)

    Remote Symmetrix ID                     : 000187900041

    R2 Device Is Larger Than The R1 Device : False

    RDF Mode                                : Synchronous
    RDF Adaptive Copy                       : Disabled
    RDF Adaptive Copy Write Pending State  : N/A
    RDF Adaptive Copy Skew (Tracks)         : 65535

    RDF Device Domino                       : Disabled
```

```
RDF Link Configuration                  : Fibre
RDF Link Domino                         : Disabled
Prevent Automatic RDF Link Recovery     : Disabled
Prevent RAs Online Upon Power ON        : Enabled

Device RDF Status                       : Ready           (RW)

Device RA Status                        : Ready           (RW)
Device Link Status                      : Not Ready       (NR)

Device Suspend State                    : Offline
Device Consistency State                : Disabled
RDF R2 Not Ready If Invalid             : Enabled

Device RDF State                        : Not Ready       (NR)
Remote Device RDF State                 : Not Ready       (NR)

RDF Pair State ( R1 <-  -> R2 )         : Suspended

Number of R1 Invalid Tracks             : 0
Number of R2 Invalid Tracks             : 0
}
```

- The following command illustrates the use of the vi text editor to create a text file named `symrep.opt`, including:
  - Parameters and values that specify the single hop configuration.
  - Define copy cycle parameters for use during the symreplicate session: one cycle with a duration of 10 minutes.

    The `CYCLE_OVERFLOW` value of `NEXT` has no relevance here (in a setup that has only one copy cycle), but is needed later when the file is edited to have two copy cycles:

```
vi symrep.opt

SYMCLI_REPLICATE_HOP_TYPE=SINGLE
SYMCLI_REPLICATE_CYCLE=10
SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT
SYMCLI_REPLICATE_NUM_CYCLES=1
```

- The `symreplicate setup` command performs the setup required to begin a replicate session.

  The difference between this command and using the `-setup` option with the `symreplicate start` command is that the latter will cycle as many times as you have specified in your `symreplicate` options file, whereas the `setup` command will cycle just once.

  If you have various “wait” options in the options file, the setup honors them.

```
symreplicate -g symrep setup -optimize -options symrep.opt -foreground -nop

Checking for valid group configuration...

Checking for valid initial group state...

Setting up local BCV pairs...

Optimizing Local BCV pairs...

Waiting for local BCV synchronization...
```

```
Splitting local BCV pairs...

Incrementally establishing RDF pairs...

Setting up remote BCV pairs...

Optimizing remote BCV pairs...

Incrementally establishing remote BCV pairs...

Waiting for remote device synchronization...

Waiting for RDF synchronization...

Splitting remote BCV pairs...

Incrementally establishing local BCV pairs...

Setup complete; exiting symreplicate...
```

- The following command runs a `symreplicate` session in the foreground so that the resulting output display illustrates the various steps involved in completing one copy cycle.

  Note: The `symrep.opt` file specified on the command line tells SYMCLI what copy cycle parameters to employ during the session. The `-consistent` option performs a consistent split of the local BCV pairs during the cycle:

```
symreplicate -g symrep start -options symrep.opt -foreground -consistent -nop

Checking for valid group configuration...

Checking for valid initial group state...

Waiting for local BCV synchronization...

Splitting local BCV pairs...

Incrementally establishing RDF pairs...

Waiting for RDF synchronization...

Suspending RDF connection...

Waiting for local BCV pairs to split...

Incrementally establishing local BCV pairs...

Incrementally establishing remote BCV pairs...

Waiting for remote device synchronization...

Splitting remote BCV pairs...

1 cycle(s) complete; exiting symreplicate...
```

- This `symreplicate` command runs one copy cycle in the background (the default when the `-foreground` option is omitted):

```
symreplicate -g symrep start -options symrep.opt -noprompt

Checking for valid group configuration...
```

```
Checking for valid initial group state...

symreplicate process launched.
```

- The `symreplicate query` command checks the status of the copy cycle being processed in the background. SYMCLI provides an updated display every five seconds. Only a representative sample of the update displays is shown below:

symreplicate -g symrep query -i 5

```
Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Waiting for next cycle             0 m        1        1

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing RDF pairs             0 m        1        1

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing local and remote      0 m        1        1

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Completed Complete                           0 m        1        1
```

- The following command uses the vi text editor again to edit the text file named `symrep.opt`.

  You can edit parameter values that affect the `symreplicate` session.

  Changing the number of copy cycles from one to two, makes the `CYCLE_OVERFLOW` value of `NEXT` relevant. If the first copy cycle lasts longer than its 10-minute time schedule, the second copy cycle will begin at the next scheduled start.

  For example, if the first copy cycle overflows to 15 minutes, the second cycle begins at the 20-minute mark. If the first copy cycle overflows to 35 minutes, the second cycle begins at the 40-minute mark:

  vi symrep.opt

  ```
  SYMCLI_REPLICATE_HOP_TYPE=SINGLE
  SYMCLI_REPLICATE_CYCLE=10
  SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT
  SYMCLI_REPLICATE_NUM_CYCLES=2
  ```

- This `symreplicate start` command runs the two-cycle session in the background:

```
symreplicate -g symrep start -options symrep.opt -noprompt

Checking for valid group configuration...

Checking for valid initial group state...
```

- The `symreplicate query` command checks the status of the copy cycles being processed in the background:

```
symreplicate -g symrep query

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ------------------------------ -------- -------- --------
SINGLE    Active    Establishing RDF pairs            10 m        1        2
```

- The `symreplicate stop` command stops the current replicate session.

  The `-step` option causes the stop to occur after the current execution step completes. Omitting `-step` would stop the session at the end of a complete copy cycle:

```
symreplicate -g symrep stop -step -noprompt

Stop operation underway.
```

- The `symreplicate query` command checks the status of the session. The display indicates that the session stopped at the step for establishing the SRDF pairs:

```
symreplicate -g symrep query

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ------------------------------ -------- -------- --------
SINGLE    Stopped   Establishing RDF pairs            10 m        1        2
```

- The `symreplicate restart` command resumes the copy cycle at the step where the session stopped.

  Specifying the options file again on `restart` is not required unless you changed the file while the session was stopped. Although the example made no changes to the options file here, the options file is specified again for consistency:

```
symreplicate -g symrep restart -options symrep.opt -noprompt

symreplicate process launched.
```

- The `symreplicate query` command checks the status of the copy cycle every five seconds and provides an updated display.

  Note: The cycle resumes where it was in the sequence of steps when the session stopped. Only a representative sample of the update displays is shown below:

```
symreplicate -g symrep query -i 5

Device Group (DG) Name     : symrep
```

```
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing RDF pairs           10 m        1        2

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing local and remote     10 m        1        2

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Waiting for next cycle            10 m        1        2

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing RDF pairs            10 m        2        2

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Active    Establishing local and remote     10 m        2        2

Device Group (DG) Name     : symrep
DG's Symmetrix ID          : 000187900035
Remote Symmetrix ID        : 000187900041

Replicate                                          Cycle    Current  Max
Hop Type  Status    Step                           Period   Cycle    Cycles
--------- --------- ----------------------------- -------- -------- --------
SINGLE    Completed Completed                         10 m        2        2
```

Example 2: SRDF/AR multi-hop configuration with BCVs at hop 2

This example is performed using Solutions Enabler version 5.4. The hardware setup consists of a Solaris host connected to a source Symmetrix (sid 79). All commands are issued from the source-side host. The example uses the following devices to create the multi-hop environment and shows how to set up the correct pair states for automated data replication in this environment:

- Source Symmetrix (sid 79): R1 devices CEE-CFD

- Hop 1 Symmetrix (sid 67): R2 devices EA2-EB1; R1 BCV devices E92-EA1
- Hop 2 Symmetrix (sid 01): R2 devices 318-327; BCV devices 328-337
- The `symrdf list` command shows the local view of the source R1 devices (SymDev), their target R2 mirror devices (RDev), and their current SRDF pair state. The ellipsis (...) represents truncated output:

```
symrdf list -r1 -sid 79

Symmetrix ID: 000187700079

                              Local Device View
---------------------------------------------------------------------------
                    STATUS   MODES                        RDF  S T A T E S
Sym         RDF    --------- -----  R1 Inv   R2 Inv ----------------------
Dev  RDev  Typ:G  SA RA LNK  MDA   Tracks   Tracks Dev RDev Pair
---- ----  ------ --------- ----- -------  ------- --- ---- -------------
0CEE 0EA2   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CEF 0EA3   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF0 0EA4   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF1 0EA5   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF2 0EA6   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF3 0EA7   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF4 0EA8   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF5 0EA9   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF6 0EAA   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF7 0EAB   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF8 0EAC   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CF9 0EAD   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CFA 0EAE   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CFB 0EAF   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CFC 0EB0   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
0CFD 0EB1   R1:64 RW RW RW   S..        0        0 RW  WD   Synchronized
...
```

- The `symdg create` command creates a device group (`symrep`). The `symdg addall` command adds the SRDF standard devices from local Symmetrix 000187700079 to the group, using the `-range` option to limit the selections to those devices between CEE and CFD. The `symbcv` command with the `-rdf` option associates a range of remote BCV devices on Hop 1 with the device group. The `symbcv` command with the `-rrdf` option associates a range of remote BCV devices on Hop 2 with the device group:

```
symdg create symrep -type rdf1
symdg -g symrep addall dev -range CEE:CFD -sid 79
symbcv -g symrep -rdf associateall dev -range E92:EA1
symbcv -g symrep -rrdf associateall dev -range 328:337
```

- The following command illustrates the use of the vi text editor to create a text file named `rep_opt.txt`. As was done here, you can enter into the file those parameters and values that specify the multi-hop configuration that uses the Hop 2 BCVs (USE_FINAL_BCV=TRUE) and define copy cycle parameters for use during the symreplicate session. The CYCLE_OVERFLOW value of NEXT has no relevance here in a setup that has only one copy cycle, but this value can play a role later if the file is edited to have more than one copy cycle:

```
vi  rep_opt.txt

SYMCLI_REPLICATE_HOP_TYPE=MULTI
SYMCLI_REPLICATE_USE_FINAL_BCV=TRUE
SYMCLI_REPLICATE_CYCLE=0
```

```
SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT
SYMCLI_REPLICATE_NUM_CYCLES=1
```

- The `symreplicate start` command with the `-setup` option sets up the required pair states, and if successful, begins the `symreplicate` session. This command will cycle as many times as you have specified in your `symreplicate` options file (the `symreplicate setup` command cycles just once):

```
symreplicate -g symrep start -setup -optimize -options rep_opt.txt -foreground
   -noprompt

Checking for valid group configuration...

Checking for valid initial group state...

Setting up local RDF pairs...

Setting up first hop BCV pairs...

Optimizing first hop BCV pairs...

Waiting for first hop BCV device synchronization...

Splitting first hop BCV pairs...

Incrementally establishing remote RDF pairs...

Setting up second hop BCV pairs...

Optimizing second hop BCV pairs...

Incrementally establishing second hop BCV pairs...

Waiting for second hop BCV synchronization...

Waiting for remote RDF pair synchronization...

Splitting second hop BCV pairs...

Incrementally establishing first hop BCV pairs...

Setup is complete...

1 cycle(s) complete; exiting symreplicate...
```

Example 3: SRDF/AR single-hop configuration using a CG

This example is performed using Solutions Enabler version 5.4.

The hardware setup consists of a single-hop configuration in which the source devices span three Symmetrix arrays (SIDs 35, 43, and 60).

A composite group is defined on a host connected to these three Symmetrix arrays.

The devices include standard devices and R1 BCV devices from the local Symmetrix arrays, as well as BCVs from the remote Symmetrix arrays.

- The `symcg create` command creates a Regular type composite group named `single-hop`:

```
symcg create single-hop -type regular
```

- The following `symcg` commands add to the composite group a range of standard devices from each of the three local source Symmetrix arrays:

```
symcg -cg single-hop addall dev -range 56:6E -sid 35
symcg -cg single-hop addall dev -range 61:79 -sid 43
symcg -cg single-hop addall dev -range 14:27 -sid 60
```

- The following `symbcv` commands associate with the composite group a range of R1 BCV devices from each of the three local source Symmetrix arrays:

```
symbcv -cg single-hop associateall dev -range 182:19A -sid 35
symbcv -cg single-hop associateall dev -range 142:15A -sid 43
symbcv -cg single-hop associateall dev -range 3B6:3C9 -sid 60
```

- The following `symbcv` commands with the `-rdf` option associate a range of BCV devices from each of the *remote* Symmetrix arrays.

  The `-bcv` option specifies that the source devices are local R1 BCV devices. If there is more than one SRDF group on a local Symmetrix array, you must include the SRDF group number of the local source devices (the group number of the R1 BCVs). Specifying the SRDF group number creates each R1/R2 pairing as well as the SRDF link for that pair:

```
symbcv -cg single-hop associateall dev -range 137:14F -bcv -rdf -sid 35 -rdfg 1
symbcv -cg single-hop associateall dev -range 12A:142 -bcv -rdf -sid 43 -rdfg 1
symbcv -cg single-hop associateall dev -range 21C:22F -bcv -rdf -sid 60 -rdfg 1
```

- The `symcg list` command displays a list of composite groups defined on this host:

```
symcg list

                    C O M P O S I T E   G R O U P S

                                          Number of          Number of
    Name                  Type    Valid  Symms   RAGs    Devs   BCVs   VDEVs

    single-hop            REGULAR Yes        3      3      64    128       0
```

- The `symcg show` command displays detailed configuration and status information about the composite group:

```
symcg show single-hop

Composite Group Name:  single-hop

  Composite Group Type                                : REGULAR
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : No

  Number of RDF (RA) Groups                           :    3
  Number of STD Devices                               :   64
  Number of BCV's (Locally-associated)                :   64
  Number of VDEV's (Locally-associated)               :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :    0
  Number of BRBCV's (Remotely-associated BCV-RDF)     :   64
  Number of RRBCV's (Remotely-associated RBCV)        :    0

  Number of Symmetrix Units (3):
     {

     1) Symmetrix ID                                    : 000187900035
        Microcode Version                               : 5670
        Number of STD Devices                           :   22
```

```
Number of BCV's (Locally-associated)           :   22
Number of VDEV's (Locally-associated)          :    0
Number of RBCV's (Remotely-associated STD_RDF) :    0
Number of BRBCV's (Remotely-associated BCV-RDF):   22
Number of RRBCV's (Remotely-associated RBCV)   :    0

Number of RDF (RA) Groups (1):
   {

   1) RDF (RA) Group Number  :  1            (00)
      Remote Symmetrix ID    : 000187900041
      Microcode Version      : 5670

      BCV's (Locally-associated) (22):
         {
         ---------------------------------------------------------------
                                 Sym  Device        Consistency   Cap
         PdevName                Dev  Config        State         (MB)
         ---------------------------------------------------------------
         N/A                     0182 RDF1-BCV+Mir  Disabled     12946
         N/A                     0185 RDF1-BCV+Mir  Disabled      8631
         N/A                     0187 RDF1-BCV+Mir  Disabled      4315
         N/A                     0188 RDF1-BCV+Mir  Disabled      4315
         N/A                     0189 RDF1-BCV+Mir  Disabled      4315
         N/A                     018A RDF1-BCV+Mir  Disabled      4315
         N/A                     018B RDF1-BCV+Mir  Disabled      4315
         N/A                     018C RDF1-BCV+Mir  Disabled      4315
         N/A                     018D RDF1-BCV+Mir  Disabled      4315
         N/A                     018E RDF1-BCV+Mir  Disabled      4315
         N/A                     018F RDF1-BCV+Mir  Disabled      4315
         N/A                     0190 RDF1-BCV+Mir  Disabled      4315
         N/A                     0191 RDF1-BCV+Mir  Disabled      4315
         N/A                     0192 RDF1-BCV+Mir  Disabled      4315
         N/A                     0193 RDF1-BCV+Mir  Disabled      4315
         N/A                     0194 RDF1-BCV+Mir  Disabled      4315
         N/A                     0195 RDF1-BCV+Mir  Disabled      4315
         N/A                     0196 RDF1-BCV+Mir  Disabled      4315
         N/A                     0197 RDF1-BCV+Mir  Disabled      4315
         N/A                     0198 RDF1-BCV+Mir  Disabled      4315
         N/A                     0199 RDF1-BCV+Mir  Disabled      4315
         N/A                     019A RDF1-BCV+Mir  Disabled      4315
         }

      BRBCV's (Remotely-associated BCV-RDF) (22):
         {
         ---------------------------------------------------------------
                                 Sym  Device        Consistency   Cap
         PdevName                Dev  Config        State         (MB)
         ---------------------------------------------------------------
         N/A                     0137 BCV           N/A          12946
         N/A                     013A BCV           N/A           8631
         N/A                     013C BCV           N/A           4315
         N/A                     013D BCV           N/A           4315
         N/A                     013E BCV           N/A           4315
         N/A                     013F BCV           N/A           4315
         N/A                     0140 BCV           N/A           4315
         N/A                     0141 BCV           N/A           4315
         N/A                     0142 BCV           N/A           4315
         N/A                     0143 BCV           N/A           4315
         N/A                     0144 BCV           N/A           4315
         N/A                     0145 BCV           N/A           4315
         N/A                     0146 BCV           N/A           4315
         N/A                     0147 BCV           N/A           4315
         N/A                     0148 BCV           N/A           4315
         N/A                     0149 BCV           N/A           4315
         N/A                     014A BCV           N/A           4315
```

```
            N/A                     014B BCV          N/A          4315
            N/A                     014C BCV          N/A          4315
            N/A                     014D BCV          N/A          4315
            N/A                     014E BCV          N/A          4315
            N/A                     014F BCV          N/A          4315
            }
         }

      STD Devices (non-RDF) (22):
         {
         ------------------------------------------------------------
                                 Sym  Device        Consistency   Cap
         PdevName                Dev  Config        State         (MB)
         ------------------------------------------------------------
         /dev/vx/rdmp/c15t2d2s2  0056 2-Way Mir     N/A          12946
         /dev/vx/rdmp/c15t2d3s2  0059 2-Way Mir     N/A           8631
         /dev/vx/rdmp/c15t2d4s2  005B 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d5s2  005C 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d6s2  005D 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d7s2  005E 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d8s2  005F 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d9s2  0060 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d10s2 0061 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d11s2 0062 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d12s2 0063 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d13s2 0064 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d14s2 0065 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d15s2 0066 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d16s2 0067 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d17s2 0068 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d18s2 0069 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d19s2 006A 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d20s2 006B 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d21s2 006C 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d22s2 006D 2-Way Mir     N/A           4315
         /dev/vx/rdmp/c15t2d23s2 006E 2-Way Mir     N/A           4315
         }

   2) Symmetrix ID                                  : 000000003143
      Microcode Version                             : 5267
      Number of STD Devices                         :   22
      Number of BCV's (Locally-associated)          :   22
      Number of VDEV's (Locally-associated)         :    0
      Number of RBCV's (Remotely-associated STD_RDF) :    0
      Number of BRBCV's (Remotely-associated BCV-RDF):   22
      Number of RRBCV's (Remotely-associated RBCV)   :    0

      Number of RDF (RA) Groups (1):
         {

         1) RDF (RA) Group Number  :  1            (A)
            Remote Symmetrix ID    : 000000003156
            Microcode Version      : 5267

            BCV's (Locally-associated) (22):
               {
               ------------------------------------------------------------
                                       Sym  Device        Consistency   Cap
               PdevName                Dev  Config        State         (MB)
               ------------------------------------------------------------
               N/A                     0142 RDF1-BCV      Disabled     12946
               N/A                     0145 RDF1-BCV      Disabled      8631
               N/A                     0147 RDF1-BCV      Disabled      4315
               N/A                     0148 RDF1-BCV      Disabled      4315
               N/A                     0149 RDF1-BCV      Disabled      4315
               N/A                     014A RDF1-BCV      Disabled      4315
```

```
            N/A                     014B RDF1-BCV      Disabled      4315
            N/A                     014C RDF1-BCV      Disabled      4315
            N/A                     014D RDF1-BCV      Disabled      4315
            N/A                     014E RDF1-BCV      Disabled      4315
            N/A                     014F RDF1-BCV      Disabled      4315
            N/A                     0150 RDF1-BCV      Disabled      4315
            N/A                     0151 RDF1-BCV      Disabled      4315
            N/A                     0152 RDF1-BCV      Disabled      4315
            N/A                     0153 RDF1-BCV      Disabled      4315
            N/A                     0154 RDF1-BCV      Disabled      4315
            N/A                     0155 RDF1-BCV      Disabled      4315
            N/A                     0156 RDF1-BCV      Disabled      4315
            N/A                     0157 RDF1-BCV      Disabled      4315
            N/A                     0158 RDF1-BCV      Disabled      4315
            N/A                     0159 RDF1-BCV      Disabled      4315
            N/A                     015A RDF1-BCV      Disabled      4315
            }

         BRBCV's (Remotely-associated BCV-RDF) (22):
            {
            ----------------------------------------------------------------
                                    Sym  Device        Consistency   Cap
            PdevName                Dev  Config        State         (MB)
            ----------------------------------------------------------------
            N/A                     012A BCV           N/A          12946
            N/A                     012D BCV           N/A           8631
            N/A                     012F BCV           N/A           4315
            N/A                     0130 BCV           N/A           4315
            N/A                     0131 BCV           N/A           4315
            N/A                     0132 BCV           N/A           4315
            N/A                     0133 BCV           N/A           4315
            N/A                     0134 BCV           N/A           4315
            N/A                     0135 BCV           N/A           4315
            N/A                     0136 BCV           N/A           4315
            N/A                     0137 BCV           N/A           4315
            N/A                     0138 BCV           N/A           4315
            N/A                     0139 BCV           N/A           4315
            N/A                     013A BCV           N/A           4315
            N/A                     013B BCV           N/A           4315
            N/A                     013C BCV           N/A           4315
            N/A                     013D BCV           N/A           4315
            N/A                     013E BCV           N/A           4315
            N/A                     013F BCV           N/A           4315
            N/A                     0140 BCV           N/A           4315
            N/A                     0141 BCV           N/A           4315
            N/A                     0142 BCV           N/A           4315
            }
      }

   STD Devices (non-RDF) (22):
      {
      ----------------------------------------------------------------
                                Sym  Device        Consistency   Cap
      PdevName                  Dev  Config        State         (MB)
      ----------------------------------------------------------------
      /dev/vx/rdmp/c15t1d2s2    0061 2-Way Mir     N/A          12946
      /dev/vx/rdmp/c15t1d3s2    0064 2-Way Mir     N/A           8631
      /dev/vx/rdmp/c15t1d4s2    0066 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d5s2    0067 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d6s2    0068 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d7s2    0069 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d8s2    006A 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d9s2    006B 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d10s2   006C 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d11s2   006D 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d12s2   006E 2-Way Mir     N/A           4315
```

```
      /dev/vx/rdmp/c15t1d13s2 006F 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d14s2 0070 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d15s2 0071 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d16s2 0072 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d17s2 0073 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d18s2 0074 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d19s2 0075 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d20s2 0076 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d21s2 0077 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d22s2 0078 2-Way Mir     N/A           4315
      /dev/vx/rdmp/c15t1d23s2 0079 2-Way Mir     N/A           4315
      }

3) Symmetrix ID                                  : 000184500160
   Microcode Version                             : 5568
   Number of STD Devices                         :   20
   Number of BCV's (Locally-associated)          :   20
   Number of VDEV's (Locally-associated)         :    0
   Number of RBCV's (Remotely-associated STD_RDF) :    0
   Number of BRBCV's (Remotely-associated BCV-RDF):   20
   Number of RRBCV's (Remotely-associated RBCV)   :    0

   Number of RDF (RA) Groups (1):
      {

      1) RDF (RA) Group Number  :  1            (A)
         Remote Symmetrix ID    : 000184502898
         Microcode Version      : 5568

         BCV's (Locally-associated) (20):
            {
            ---------------------------------------------------------------
                                    Sym  Device        Consistency   Cap
            PdevName                Dev  Config        State         (MB)
            ---------------------------------------------------------------
            N/A                     03B6 RDF1-BCV+Mir  Disabled      4315
            N/A                     03B7 RDF1-BCV+Mir  Disabled      4315
            N/A                     03B8 RDF1-BCV+Mir  Disabled      4315
            N/A                     03B9 RDF1-BCV+Mir  Disabled      4315
            N/A                     03BA RDF1-BCV+Mir  Disabled      4315
            N/A                     03BB RDF1-BCV+Mir  Disabled      4315
            N/A                     03BC RDF1-BCV+Mir  Disabled      4315
            N/A                     03BD RDF1-BCV+Mir  Disabled      4315
            N/A                     03BE RDF1-BCV+Mir  Disabled      4315
            N/A                     03BF RDF1-BCV+Mir  Disabled      4315
            N/A                     03C0 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C1 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C2 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C3 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C4 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C5 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C6 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C7 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C8 RDF1-BCV+Mir  Disabled      4315
            N/A                     03C9 RDF1-BCV+Mir  Disabled      4315
            }

         BRBCV's (Remotely-associated BCV-RDF) (20):
            {
            ---------------------------------------------------------------
                                    Sym  Device        Consistency   Cap
            PdevName                Dev  Config        State         (MB)
            ---------------------------------------------------------------
            N/A                     021C 2-Way BCV Mir N/A           4315
            N/A                     021D 2-Way BCV Mir N/A           4315
            N/A                     021E 2-Way BCV Mir N/A           4315
```

```
            N/A                     021F 2-Way BCV Mir N/A          4315
            N/A                     0220 2-Way BCV Mir N/A          4315
            N/A                     0221 2-Way BCV Mir N/A          4315
            N/A                     0222 2-Way BCV Mir N/A          4315
            N/A                     0223 2-Way BCV Mir N/A          4315
            N/A                     0224 2-Way BCV Mir N/A          4315
            N/A                     0225 2-Way BCV Mir N/A          4315
            N/A                     0226 2-Way BCV Mir N/A          4315
            N/A                     0227 2-Way BCV Mir N/A          4315
            N/A                     0228 2-Way BCV Mir N/A          4315
            N/A                     0229 2-Way BCV Mir N/A          4315
            N/A                     022A 2-Way BCV Mir N/A          4315
            N/A                     022B 2-Way BCV Mir N/A          4315
            N/A                     022C 2-Way BCV Mir N/A          4315
            N/A                     022D 2-Way BCV Mir N/A          4315
            N/A                     022E 2-Way BCV Mir N/A          4315
            N/A                     022F 2-Way BCV Mir N/A          4315
            }
         }

      STD Devices (non-RDF) (20):
         {
         ----------------------------------------------------------------
                                  Sym  Device        Consistency   Cap
         PdevName                 Dev  Config        State         (MB)
         ----------------------------------------------------------------
         /dev/rdsk/emcpower38c    0014 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower39c    0015 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower40c    0016 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower41c    0017 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower42c    0018 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower43c    0019 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower44c    001A 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower45c    001B 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower46c    001C 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower47c    001D 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower48c    001E 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower49c    001F 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower50c    0020 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower51c    0021 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower52c    0022 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower53c    0023 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower54c    0024 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower55c    0025 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower56c    0026 2-Way Mir     N/A           4315
         /dev/rdsk/emcpower57c    0027 2-Way Mir     N/A           4315
         }
}
```

- The following command illustrates the use of the vi text editor to create a text file named `sar.opt`.

  The setup operation requires that the `HOP_TYPE` be defined. You define copy cycle parameters for use during the `symreplicate` session (for example, two cycles where the time from the beginning of the first cycle to the beginning of the next cycle should be 10 minutes):

  vi sar.opt

  ```
  SYMCLI_REPLICATE_HOP_TYPE=SINGLE
  SYMCLI_REPLICATE_CYCLE=10
  SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT
  SYMCLI_REPLICATE_NUM_CYCLES=2
  ```

- Similar to "Example 1: SRDF/AR single-hop configuration", the `symreplicate setup` command performs the setup here for all pairs in the composite group that spans three source Symmetrix arrays.

  This command results in the setup performing one cycle:

```
symreplicate -cg single-hop setup -foreground -options sar.opt -noprompt

Checking for valid group configuration...

Checking for valid initial group state...

Setting up local BCV pairs...

Waiting for local BCV synchronization...

Splitting local BCV pairs...

Incrementally establishing RDF pairs...

Setting up remote BCV pairs...

Incrementally establishing remote BCV pairs...

Waiting for remote device synchronization...

Waiting for RDF synchronization...

Splitting remote BCV pairs...

Incrementally establishing local BCV pairs...

Setup complete; exiting symreplicate...
```

- The setup is complete. You can now perform `symreplicate start` for the composite group as was done for the device group in Example 1.

Example 4: SRDF/AR multi-hop configuration using a CG

This example is performed using Solutions Enabler version 5.4.

The hardware setup consists of a multi-hop configuration in which the source devices span two Symmetrix arrays (SIDs 79 and 32).

A composite group is defined on a Solaris host connected to these two Symmetrix arrays.

The devices include R1 devices from the local Symmetrix arrays, as well as BCVs from on the remote Hop-1 and Hop-2 Symmetrix arrays.

- The `symcg create` command creates an RDF1 type composite group named `multi-hop`:

```
symcg create multi-hop -type rdf1
```

- The following `symcg` commands add to the composite group a range of standard devices from each of the two local source Symmetrix arrays:

```
symcg -cg multi-hop addall dev -range CEE:CFD -sid 79
symcg -cg multi-hop addall dev -range 2:7 -sid 32
```

- The following `symbcv` commands with the `-rdf` option associate a range of BCV devices from each of the *remote* Hop-1 Symmetrix arrays.

If there is more than one SRDF group on a local Symmetrix array, you must include the SRDF group number of the local R1 source devices. Specifying the SRDF group number creates the SRDF link for each R1/R2 pair:

```
symbcv -cg multi-hop associateall dev -range E92:EA1 -rdf -rdfg 64 -sid 79
symbcv -cg multi-hop associateall dev -range 48:4D -rdf -rdfg 1 -sid 32
```

- The following `symbcv` commands with the `-rrdf` option associate a range of BCV devices from each of the *remote* Hop-2 Symmetrix arrays. To define the path of the SRDF link, you must always include the group number of the local source devices as specified with the `-rdfg` option:[1]

```
symbcv -cg multi-hop associateall dev -range 328:337 -rrdf -rdfg 64 -sid 79
symbcv -cg multi-hop associateall dev -range 30:35 -rrdf -rdfg 1 -sid 32
```

- The `symcg list` command displays a list of composite groups defined on this host:

```
symcg list

                    C O M P O S I T E   G R O U P S

                                          Number of          Number of
   Name                  Type    Valid  Symms   RAGs    Devs   BCVs   VDEVs

   multi-hop             RDF1    Yes        2      2      22     44       0
```

- The `symcg show` command displays configuration and status information about the composite group:

```
symcg show multi-hop

Composite Group Name:  multi-hop

  Composite Group Type                                : RDF1
  Valid                                               : Yes
  CG in PowerPath                                     : No
  CG in GNS                                           : No

  Number of RDF (RA) Groups                           :    2
  Number of STD Devices                               :   22
  Number of BCV's (Locally-associated)                :    0
  Number of VDEV's (Locally-associated)               :    0
  Number of RBCV's (Remotely-associated STD-RDF)      :   22
  Number of BRBCV's (Remotely-associated BCV-RDF)     :    0
  Number of RRBCV's (Remotely-associated RBCV)        :   22

  Number of Symmetrix Units (2):
     {

     1) Symmetrix ID                                   : 000187700079
        Microcode Version                              : 5670
        Number of STD Devices                          :   16
        Number of BCV's (Locally-associated)           :    0
        Number of VDEV's (Locally-associated)          :    0
        Number of RBCV's (Remotely-associated STD_RDF) :   16
        Number of BRBCV's (Remotely-associated BCV-RDF):    0
        Number of RRBCV's (Remotely-associated RBCV)   :   16

        Number of RDF (RA) Groups (1):
           {
```

1. If you add the Hop 2 BCVs to the composite group before adding the Hop 1 BCVs, include the `-remote_rdfg` option also.

```
1) RDF (RA) Group Number  : 64            (3F)
   Remote Symmetrix ID    : 000187700067
   Microcode Version      : 5670

   STD Devices (16):
      {
      ------------------------------------------------------------
                              Sym  Device        Consistency   Cap
      PdevName                Dev  Config        State         (MB)
      ------------------------------------------------------------
      /dev/vx/rdmp/c15t0d63s2 0CEE RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d64s2 0CEF RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d65s2 0CF0 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d66s2 0CF1 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d67s2 0CF2 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d68s2 0CF3 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d69s2 0CF4 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d70s2 0CF5 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d71s2 0CF6 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d72s2 0CF7 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d73s2 0CF8 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d74s2 0CF9 RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d75s2 0CFA RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d76s2 0CFB RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d77s2 0CFC RDF1+Mir      Disabled      4315
      /dev/vx/rdmp/c15t0d78s2 0CFD RDF1+Mir      Disabled      4315
      }

   RBCV's (Remotely-associated STD-RDF) (16):
      {
      Remote RDF (RA) Group Number  : 52            (33)
      Remote Remote Symmetrix ID    : 000000006201
      Microcode Version             : 5670

      ------------------------------------------------------------
                              Sym  Device        Consistency   Cap
      PdevName                Dev  Config        State         (MB)
      ------------------------------------------------------------
      N/A                     0E92 RDF1-BCV      Disabled      4315
      N/A                     0E93 RDF1-BCV      Disabled      4315
      N/A                     0E94 RDF1-BCV      Disabled      4315
      N/A                     0E95 RDF1-BCV      Disabled      4315
      N/A                     0E96 RDF1-BCV      Disabled      4315
      N/A                     0E97 RDF1-BCV      Disabled      4315
      N/A                     0E98 RDF1-BCV      Disabled      4315
      N/A                     0E99 RDF1-BCV      Disabled      4315
      N/A                     0E9A RDF1-BCV      Disabled      4315
      N/A                     0E9B RDF1-BCV      Disabled      4315
      N/A                     0E9C RDF1-BCV      Disabled      4315
      N/A                     0E9D RDF1-BCV      Disabled      4315
      N/A                     0E9E RDF1-BCV      Disabled      4315
      N/A                     0E9F RDF1-BCV      Disabled      4315
      N/A                     0EA0 RDF1-BCV      Disabled      4315
      N/A                     0EA1 RDF1-BCV      Disabled      4315
      }

   RRBCV's (Remotely-associated RBCV) (16):
      {
      ------------------------------------------------------------
                              Sym  Device        Consistency   Cap
      PdevName                Dev  Config        State         (MB)
      ------------------------------------------------------------
      N/A                     0328 BCV           N/A           4315
      N/A                     0329 BCV           N/A           4315
      N/A                     032A BCV           N/A           4315
```

```
            N/A                     032B BCV           N/A           4315
            N/A                     032C BCV           N/A           4315
            N/A                     032D BCV           N/A           4315
            N/A                     032E BCV           N/A           4315
            N/A                     032F BCV           N/A           4315
            N/A                     0330 BCV           N/A           4315
            N/A                     0331 BCV           N/A           4315
            N/A                     0332 BCV           N/A           4315
            N/A                     0333 BCV           N/A           4315
            N/A                     0334 BCV           N/A           4315
            N/A                     0335 BCV           N/A           4315
            N/A                     0336 BCV           N/A           4315
            N/A                     0337 BCV           N/A           4315
            }
      }

2) Symmetrix ID                                  : 000000005232
   Microcode Version                             : 5568
   Number of STD Devices                         :    6
   Number of BCV's (Locally-associated)          :    0
   Number of VDEV's (Locally-associated)         :    0
   Number of RBCV's (Remotely-associated STD_RDF) :    6
   Number of BRBCV's (Remotely-associated BCV-RDF):    0
   Number of RRBCV's (Remotely-associated RBCV)   :    6

   Number of RDF (RA) Groups (1):
      {

      1) RDF (RA) Group Number  :  1            (A)
         Remote Symmetrix ID    : 000000005231
         Microcode Version      : 5568

         STD Devices (6):
            {
            ---------------------------------------------------------------
                                    Sym  Device        Consistency   Cap
            PdevName                Dev  Config        State         (MB)
            ---------------------------------------------------------------
            N/A                     0002 RDF1          Disabled      1031
            N/A                     0003 RDF1          Disabled      1031
            N/A                     0004 RDF1          Disabled      1031
            N/A                     0005 RDF1          Disabled      1031
            N/A                     0006 RDF1          Disabled      1031
            N/A                     0007 RDF1          Disabled      1031
            }

         RBCV's (Remotely-associated STD-RDF) (6):
            {
            Remote RDF (RA) Group Number  :  2            (B)
            Remote Remote Symmetrix ID    : 000000005233
            Microcode Version             : 5568

            ---------------------------------------------------------------
                                    Sym  Device        Consistency   Cap
            PdevName                Dev  Config        State         (MB)
            ---------------------------------------------------------------
            N/A                     0048 RDF1-BCV      Disabled      1031
            N/A                     0049 RDF1-BCV      Disabled      1031
            N/A                     004A RDF1-BCV      Disabled      1031
            N/A                     004B RDF1-BCV      Disabled      1031
            N/A                     004C RDF1-BCV      Disabled      1031
            N/A                     004D RDF1-BCV      Disabled      1031
            }

         RRBCV's (Remotely-associated RBCV) (6):
            {
```

```
                    ----------------------------------------------------------------
                                                Sym  Device        Consistency   Cap
                    PdevName                    Dev  Config        State         (MB)
                    ----------------------------------------------------------------
                    N/A                         0030 BCV           N/A           1031
                    N/A                         0031 BCV           N/A           1031
                    N/A                         0032 BCV           N/A           1031
                    N/A                         0033 BCV           N/A           1031
                    N/A                         0034 BCV           N/A           1031
                    N/A                         0035 BCV           N/A           1031
                    }
            }
}
```

- The following command illustrates the use of the vi text editor to create a text file named `3-hop.opt`. The setup operation requires that the `HOP_TYPE` be defined. By default, `USE_FINAL_BCV` is set to `TRUE`. You define copy cycle parameters for use during the `symreplicate` session (for example, two cycles where the time from the beginning of the first cycle to the beginning of the next cycle should be 10 minutes):

  vi 3-hop_opt.txt

  ```
  SYMCLI_REPLICATE_HOP_TYPE=MULTI
  SYMCLI_REPLICATE_USE_FINAL_BCV=TRUE
  SYMCLI_REPLICATE_CYCLE=10
  SYMCLI_REPLICATE_CYCLE_OVERFLOW=NEXT
  SYMCLI_REPLICATE_NUM_CYCLES=2
  ```

- Similar to Example 1, the `symreplicate setup` command performs the setup here for all pairs in the composite group that spans two source Symmetrix arrays. This command results in the setup performing one cycle. The `-optimize` flag is included to optimize BCV device pairings within each Symmetrix array:

```
symreplicate -cg multi-hop setup -optimize -foreground -options 3-hop.opt -nop
Checking for valid group configuration...
Checking for valid initial group state...
Setting up local RDF pairs...
Setting up first hop BCV pairs...
Optimizing first hop BCV pairs...
Waiting for first hop BCV device synchronization...
Splitting first hop BCV pairs...
Incrementally establishing remote RDF pairs...
Setting up second hop BCV pairs...
Optimizing second hop BCV pairs...
Incrementally establishing second hop BCV pairs...
Waiting for second hop BCV synchronization...
Waiting for remote RDF pair synchronization...
Splitting second hop BCV pairs...
Incrementally establishing first hop BCV pairs...
Setup complete; exiting symreplicate...
```

- The setup is complete. You can now perform `symreplicate start` for the composite group as was done for the device group in Example 2.

Example 5: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (single-hop configuration)

The hardware setup for this single-hop SRDF/AR configuration consists of a host connected to a local source Symmetrix (sid 505) and a remote target Symmetrix. All commands are issued from the source-side host and affect devices on the local Symmetrix. The example establishes a set of non-SRDF/AR BCVs that are concurrent with a set of SRDF/AR BCVs. Devices on the local Symmetrix are:

- SRDF/AR standard devices 012E and 012F
- SRDF/AR BCV devices 04CE and 04CF
- Non-SRDF/AR BCV devices 04CC and 04CD
- The SRDF/AR device group (`sar`) and the creation of the single-hop environment have already been set up. The SRDF/AR device group was created using the following commands:

```
symdg create sar
symdg -g sar addall dev -range 12E:12F
symbcv -g sar associateall dev -range 4CE:4CF
symbcv -g sar associateall dev -range 30C:30D -bcv -rdf
```

- A device file named `devfile` was created to define the following non-SRDF/AR BCV pairs:

  ```
  012E 04CC
  012F 04CD
  ```

- The following commands were used to fully establish and split the non-SRDF/AR BCV pairs in the device file:

  ```
  symmir -f devfile -sid 505 establish -full -noprompt
  symmir -f devfile -sid 505 split -noprompt
  ```

- The SRDF/AR devices were set up for an SRDF/AR copy cycle using the following commands that operate on the SRDF/AR devices in the device group named `sar`. (Beginning with Solutions Enabler version 5.4, you can perform these steps automatically with the `symreplicate setup` command.)

  ```
  symmir -g sar establish -full -exact -noprompt
  symmir -g sar split -noprompt
  symrdf -g sar -bcv establish -noprompt
  symrdf -g sar -bcv split -noprompt
  symmir -g sar establish -full -exact -bcv -rdf -noprompt
  symmir -g sar split -bcv -rdf -noprompt
  symmir -g sar establish -noprompt
  ```

  If a concurrent BCV setup exists when an SRDF/AR copy cycle begins, SRDF/AR will use the last BCV pair that was established, regardless of whether it was the SRDF/AR BCV pair or the non-SRDF/AR BCV pair. *Prior to starting SRDF/AR, you need to make sure that the last BCV pairs that were established were the SRDF/AR BCV pairs*. In the preceding sequence, the last BCV pairs established were the SRDF/AR BCV pairs. Thus, the required setup sequence has been performed.

- The following `symmir query` command with the `-multi` option examines the relationship and status of the concurrent BCV pairs. The SRDF/AR pairs are devices 012E/04CE and 012F/04CF. The non-SRDF/AR pairs are devices 012E/04CC and 012F/04CD. The non-SRDF/AR pairs are displayed even though they are *not* associated with the device group (indicated by the **N/A** and the absence of the *). All pairs are in the **Synchronized** state:

```
symmir -g sar query -multi

Device Group (DG) Name: sar
DG's Type             : REGULAR
DG's Symmetrix ID     : 000185500505

     Standard Device                    BCV Device                  State
------------------------- ------------------------------------- ----------
                     Inv.                                  Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

DEV001         012E      0 BCV001               04CE *         0 Synchronized
                         0 N/A                  04CC           0 Synchronized
DEV002         012F      0 BCV002               04CF *         0 Synchronized
                         0 N/A                  04CD           0 Synchronized

Total                ------                                 -------
  Track(s)                0                                      0
  MB(s)                 0.0                                    0.0

Legend:
```

```
(*): The paired BCV device is associated with this group.
```

- The `symreplicate` command runs one SRDF/AR copy cycle in the foreground, using the configuration of devices defined in the device group `sar` and single-hop copy options defined in a file called `sar.opt` (for file content, refer to the `symrep.opt` file defined in Example 1):

```
symreplicate -g sar -options sar.opt -foreground start -noprompt

Execute a symreplicate 'Start' operation
for device group 'sar' (y/[n]) ? y

Checking for valid group configuration...

Checking for valid initial group state...

Checking for local BCV synchronization...

Splitting local BCV pairs...
```

- While the SRDF/AR copy cycle is in progress, issuing a `symmir query` command with the `-multi` option from a second window indicates that the local SRDF/AR BCV pairs are now **Split** (the preceding `symreplicate` output shows them in process of splitting). The non-SRDF/AR BCV pairs remain in the **Synchronized** state:

```
symmir -g sar query -multi

Device Group (DG) Name: sar
DG's Type             : REGULAR
DG's Symmetrix ID     : 000185500505

     Standard Device                   BCV Device                 State
------------------------- ------------------------------------- -----------
                     Inv.                                Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

DEV001         012E      0 N/A                  04CC           0 Synchronized
                         0 BCV001               04CE *        30 Split
DEV002         012F      0 N/A                  04CD           0 Synchronized
                         0 BCV002               04CF *        30 Split

Total               ------                                 -------
  Track(s)               0                                      60
  MB(s)                0.0                                     1.9

Legend:

(*): The paired BCV device is associated with this group.
```

- The `symmir split` command (from the second window) attempts to split the non-SRDF/AR BCV pairs that were defined in the device file named devfile. However, without the `-skip` option, this operation fails because the standard devices are locked as a result of their participation in the SRDF/AR copy cycle:

```
symmir -f devfile -sid 505 split -instant -noprompt

'Split' operation execution is in progress for the device list in
   device file
   'devfile'. Please wait...

Unable to acquire the Symmetrix device lock
```

- Another attempt to split the non-SRDF/AR BCV pairs uses the `-skip` option and is successful:

```
symmir -f devfile -sid 505 split -instant -noprompt -skip

'Split' operation execution is in progress for the device list in
   device file
   'devfile'. Please wait...

'Split' operation successfully executed for the device list in device
   file
   'devfile'.
```

- Another query shows that both the SRDF/AR and non-SRDF/AR BCV pairs are in the **Split** state:

```
symmir -g sar query -multi

Device Group (DG) Name: sar
DG's Type             : REGULAR
DG's Symmetrix ID     : 000185500505

     Standard Device                   BCV Device                 State
------------------------- ------------------------------------- -----------
                    Inv.                                 Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

DEV001         012E      0 N/A                  04CC          30 Split
                         0 BCV001               04CE *        30 Split
DEV002         012F      0 N/A                  04CD          30 Split
                         0 BCV002               04CF *        30 Split

Total               ------                               -------
  Track(s)               0                                   120
  MB(s)                0.0                                   3.8

Legend:

(*): The paired BCV device is associated with this group.
```

Note: The following output in the first window displays the completion of the `symreplicate copy` cycle that began earlier. Note that the local SRDF/AR BCV pairs are re-established prior to completion.

```
Incrementally establishing RDF pairs...

Waiting for RDF synchronization...

Incrementally establishing local BCV pairs...

Incrementally establishing remote BCV pairs...

Waiting for remote device synchronization...

Splitting remote BCV pairs...

1 cycle(s) complete; exiting symreplicate...
```

Example 6: Accessing concurrent non-SRDF/AR BCVs while running SRDF/AR (multi-hop configuration)

The hardware setup for this multi-hop SRDF/AR configuration consists of two hosts — one connected to a local (source) Symmetrix, and the other connected to a remote (target) Symmetrix (sid 33) at Hop 2. Some commands are issued from the local-site host and some from the remote-site host. The SRDF/AR device group (`sar`) and the creation of the multi-hop environment have already been set up. The example establishes on the target Symmetrix a set of non-SRDF/AR BCVs that are concurrent with a set of SRDF/AR BCVs there. Devices on the target (Hop 2) Symmetrix are:

- SRDF/AR standard devices 0001–0005
- SRDF/AR BCV devices 0031–0035
- Non-SRDF/AR BCV devices 0043–0047

Although the local-site host has a device *group* defined for running SRDF/AR, you also need to create a device *file* that allows the local-site host to manipulate the SRDF/AR BCV pairs located on the remote Symmetrix array (sid 33) at Hop 2. The following `symmir query` command from the local-site host examines the status of the SRDF/AR BCV pairs on Hop 2 that were defined previously in device file `devfile`:[1]

```
symmir -f devfile query -sid 33

Device File Name      : devfile
Device's Symmetrix ID : 000000005233

     Standard Device                    BCV Device                  State
------------------------- ------------------------------------- -----------
                     Inv.                                  Inv.
Logical         Sym  Tracks Logical               Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

N/A             0001      0 N/A                   0031           0 Split
N/A             0002      0 N/A                   0032           0 Split
N/A             0003      0 N/A                   0033           0 Split
N/A             0004      0 N/A                   0034           0 Split
N/A             0005      0 N/A                   0035           0 Split

Total               -------                                 -------
  Track(s)                0                                       0
  MB(s)                 0.0                                     0.0

Legend:

(*): The paired BCV device is associated with this group.
```

- On the remote-site host, create a device group to manipulate the non-SRDF/AR BCVs. The `symdg create` command creates an R2 type device group (`mbcv`). The `symdg addall` command adds the SRDF/AR R2 devices to the group, using the `-range` option to limit the selections to those devices between 0001 and 0005. The `symbcv`

1. The device file devfile defines the following SRDF/AR pairs:
   0001 0031
   0002 0032
   0003 0033
   0004 0034
   0005 0035

command associates the non-SRDF/AR BCVs with the device group, using the `-range` option with the `associateall` action to limit the selection to those BCVs that are within the specified range (0043 through 0047):

```
symdg create mbcv -type rdf2
symdg-g mbcv addall dev -range 0001:0005 -sid 33
symbcv -g mbcv associateall dev -range 0043:0047
```

- The `symmir establish` command from the remote host fully establishes the BCV pairs in the exact order that they were defined in the device group:

```
symmir -g mbcv establish -full -exact -noprompt

'Full Establish' operation execution is in progress for device group 'mbcv'.
   Please wait...

'Full Establish' operation successfully initiated for device group 'mbcv'.
```

- The `symmir query` command from the remote host displays the status of the non-SRDF/AR BCV pairs (**SyncInProg**):

```
symmir -g mbcv query

Device Group (DG) Name: mbcv
DG's Type             : RDF2
DG's Symmetrix ID     : 000000005233

     Standard Device                   BCV Device                 State
------------------------- ------------------------------------- -----------
                     Inv.                                  Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

DEV001         0001      0 BCV001               0043 *     18740 SyncInProg
DEV002         0002      0 BCV002               0044 *      1467 SyncInProg
DEV003         0003      0 BCV003               0045 *      7614 SyncInProg
DEV004         0004      0 BCV004               0046 *      2305 SyncInProg
DEV005         0005      0 BCV005               0047 *     18913 SyncInProg

Total              -------                                -------
  Track(s)               0                                  49039
  MB(s)                0.0                                 1532.5

Legend:

(*): The paired BCV device is associated with this group.
```

- The following `symmir query` command from the remote host with the `-multi` option displays the status of the concurrent BCV pairs — both the SRDF/AR BCV pairs and the non-SRDF/AR BCV pairs. The first non-SRDF/AR pair in the display is 0001/0043; the first SRDF/AR pair is 0001/0031. The **N/A** and the absence of an asterisk (*) indicates that the SRDF/AR BCVs are not associated with this device group (`mbcv`):

```
symmir -g mbcv query -multi

Device Group (DG) Name: mbcv
DG's Type             : RDF2
DG's Symmetrix ID     : 000000005233

     Standard Device                   BCV Device                 State
------------------------- ------------------------------------- -----------
                     Inv.                                  Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------
```

```
DEV001         0001      0 BCV001               0043 *     17027 SyncInProg
                         2 N/A                  0031          0 Split
DEV002         0002      0 BCV002               0044 *         0 Synchronized
                         2 N/A                  0032          0 Split
DEV003         0003      0 BCV003               0045 *      5006 SyncInProg
                         2 N/A                  0033          0 Split
DEV004         0004      0 BCV004               0046 *         0 Synchronized
                         2 N/A                  0034          0 Split
DEV005         0005      0 BCV005               0047 *     17850 SyncInProg
                         2 N/A                  0035          0 Split

Total               ------                              -------
  Track(s)              10                                39883
  MB(s)                0.3                               1246.3

Legend:

(*): The paired BCV device is associated with this group.
```

- The `symmir verify` command from the remote host checks the state of the BCV pairs in the device group every five seconds until the non-SRDF/AR BCV pairs are synchronized:

```
symmir -g mbcv verify -i 5

Not all devices in group 'mbcv' are in the 'Synchronized or Restored' state.

Not all devices in group 'mbcv' are in the 'Synchronized or Restored' state.

All devices in group 'mbcv' are in the 'Synchronized or Restored' state.
```

- The `symmir split` command from the remote host splits the non-SRDF/AR BCV pairs:

```
symmir -g mbcv split -noprompt

'Split' operation execution is in progress for device group 'mbcv'.
   Please wait...

'Split' operation successfully executed for device group 'mbcv'.
```

- If a concurrent BCV setup exists when an SRDF/AR copy cycle begins, SRDF/AR will use the last BCV pair that was established and split — regardless of whether it was the SRDF/AR BCV pair or the non-SRDF/AR BCV pair. At this point, the last manipulated BCV pairs were the non-SRDF/AR BCV pairs. The following examples show how to establish and split the SRDF/AR BCV pairs so that they are the last pairs to be manipulated before beginning the `symreplicate copy` cycle. This manipulation is only required prior to starting SRDF/AR. The following `symmir establish` command from the local-site host establishes the SRDF/AR BCV pairs in device file `devfile`:

```
symmir -f devfile establish -noprompt -sid 33

'Incremental Establish' operation execution is in progress for the device list
   in device file 'devfile'. Please wait...

'Incremental Establish' operation successfully initiated for the device list
   in device file 'devfile'.
```

- The `symmir split` command from the local host splits the SRDF/AR BCV pairs in device file `devfile`:

```
symmir -f devfile split -nop -sid 33
```

```
'Split' operation execution is in progress for the device list in device file
   'devfile'. Please wait...

'Split' operation successfully executed for the device list in device file
   'devfile'.
```

- The `symreplicate` command runs one SRDF/AR copy cycle in the foreground from the local host, using the configuration of devices defined in the device group `sar` and multi-hop copy options defined in a file called `rep.txt` (for file content, refer to the `rep_opt.txt` file defined in "Example 2: SRDF/AR multi-hop configuration with BCVs at hop 2" on page 433):

```
symreplicate -g sar start -options rep.txt -foreground -noprompt

Checking for valid group configuration...

Checking for valid initial group state...

Checking for first hop BCV device synchronization...

Splitting first hop BCV pairs...

Incrementally establishing remote RDF pairs...

Waiting for remote RDF pair synchronization...

Incrementally establishing first hop BCV pairs...

Incrementally establishing second hop BCV pairs...

Waiting for second hop BCV synchronization...

Splitting second hop BCV pairs...
```

- While the SRDF/AR copy cycle is in progress (the preceding `symreplicate` output shows that the SRDF/AR second-hop BCV pairs are in process of splitting), the following `symmir establish` command from the remote-site host attempts to establish the non-SRDF/AR BCV pairs. However, the standard devices are locked as a result of their participation in the SRDF/AR copy cycle, so the operation fails:

```
symmir -g mbcv establish

Execute 'Incremental Establish' operation for device group 'mbcv' (y/[n]) ? y

'Incremental Establish' operation execution is in progress for device group
   'mbcv'. Please wait...

Unable to acquire the Symmetrix device lock
```

- A subsequent `symmir establish` command from the remote host uses the `-skip` option to override the device lock, thus allowing the non-SRDF/AR BCVs to be established with the SRDF/AR standard devices that are participating in the SRDF/AR copy cycle:

```
symmir -g mbcv establish -noprompt -skip

'Incremental Establish' operation execution is in progress for device group
   'mbcv'. Please wait...

'Incremental Establish' operation successfully initiated for device group
   'mbcv'.
```

- The `symmir query` command from the remote host with the `-multi` option displays the status of the concurrent BCV pairs — both the SRDF/AR BCV pairs and the non-SRDF/AR BCV pairs. The SRDF/AR BCV pairs have reached the **Split** state; the non-SRDF/AR BCV pairs are in the **Synchronized** state:

```
symmir -g mbcv query -multi

Device Group (DG) Name: mbcv
DG's Type             : RDF2
DG's Symmetrix ID     : 000000005233

     Standard Device                   BCV Device                 State
------------------------- ------------------------------------- -----------
                    Inv.                                 Inv.
Logical        Sym  Tracks Logical              Sym       Tracks STD <=> BCV
------------------------- ------------------------------------- -----------

DEV001         0001      0 BCV001               0043 *         0 Synchronized
                         0 N/A                  0031           0 Split
DEV002         0002      0 BCV002               0044 *         0 Synchronized
                         0 N/A                  0032           0 Split
DEV003         0003      0 BCV003               0045 *         0 Synchronized
                         0 N/A                  0033           0 Split
DEV004         0004      0 BCV004               0046 *         0 Synchronized
                         0 N/A                  0034           0 Split
DEV005         0005      0 BCV005               0047 *         0 Synchronized
                         0 N/A                  0035           0 Split

Total               ------                                 -------
  Track(s)               0                                       0
  MB(s)                0.0                                     0.0

Legend:

(*): The paired BCV device is associated with this group.
```

- Because the `symreplicate` copy cycle is still in progress, the following attempt from the remote host to split the non-SRDF/AR BCV pairs without the `-skip` option fails:

```
symmir -g mbcv split -noprompt

'Split' operation execution is in progress for device group 'mbcv'. Please wait...

Unable to acquire the Symmetrix device lock
```

- Another attempt from the remote host to split the non-SRDF/AR BCV pairs using the `-skip` option succeeds:

```
symmir -g mbcv split -noprompt -skip

'Split' operation execution is in progress for device group 'mbcv'. Please wait...

'Split' operation successfully executed for device group 'mbcv'.
```

Example 7: Restarting a replicate session when devices are locked

Device locks are held during the replicate session to block other applications from altering device states while this session executes. Under certain circumstances, a replicate session may exit with devices left in a locked state. For example, a replicate session may terminate when an SRDF link goes down unexpectedly. Then the replicate session cannot restart after the SRDF link is brought back up, because of the locked devices. You can use the `-recover` option with the `symreplicate start` or `restart` command to recover the existing device locks and restart the session (SRDF/AR checks if it previously owned the device locks and, if so, proceeds as if it just acquired the existing locks).

Using the `-recover` option allows you to recover without having to manually release the device locks. When SRDF/AR detects a situation where devices are locked and recovery is possible, SRDF/AR returns a message suggesting that you attempt to recover.

IMPORTANT

Caution: Before using the `-recover` option, make sure no other replicate session that uses the same device group is currently running.

- The following `symreplicate restart` command attempts to restart a replicate session involving devices in the device group `sar2`. However, the output indicates that SRDF/AR is unable to do so at this time, because it cannot “lock the local devices,” indicating that the devices in `sar2` are already in a locked state:

```
symreplicate restart -g sar2 -foreground -noprompt

Checking for valid group configuration...

Checking for valid initial group state...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices; waiting for retry...

Can't lock local devices.

Unable to acquire the Symmetrix device lock

If you are sure no other symreplicate process is currently active for
   group
'sar2', the locks can be recovered by specifying the '-recover' option.
```

The “If you are sure” message above will not be displayed if SRDF/AR detects that the locks cannot be recovered, or that the base daemon is running. If the base daemon is running, the device locks will be released eventually. You can wait a short time and retry the operation. If the base daemon is not running, you can release the device locks manually (for example, `symdg -lock release sar2`).

- The `symreplicate restart` command is repeated here using the `-recover` option. SRDF/AR resolves the locked device situation and is able to restart the replicate session normally:

```
symreplicate restart -g sar2 -foreground -noprompt -recover

Checking for valid group configuration...

Checking for valid initial group state...

Checking for first hop BCV device synchronization...

Splitting first hop BCV pairs...

Incrementally establishing remote RDF pairs...

Waiting for remote RDF pair synchronization...

Incrementally establishing first hop BCV pairs...

Incrementally establishing second hop BCV pairs...

Waiting for second hop BCV synchronization...

Splitting second hop BCV pairs...

1 cycle(s) complete; exiting symreplicate...
```

APPENDIX A
TimeFinder Snap and Clone Reference

This appendix provides reference information for SRDF operations for TimeFinder sessions. Topics include:

- SRDF operations for TimeFinder/Snap and VP Snap sessions 460
- SRDF operations for TimeFinder/Clone sessions 469
- SRDF operations for Extent-level TimeFinder/Clone sessions 480
- SRDF set operations for TimeFinder/Snap sessions 490
- SRDF set operations for TimeFinder/Clone sessions 492

SRDF operations for TimeFinder/Snap and VP Snap sessions

This section lists the allowable SRDF operations for TimeFinder/Snap and TimeFinder VP Snap copy sessions on the R1 source and target and the R2 source and target.

Note: TimeFinder/Snap and TimeFinder VP Snap are separate features. They are combined in this section because their interactions with SRDF are very similar.

Some footnotes in the tables below refer to devices that are not pace-capable. For additional details, see "Identifying devices that cannot be paced in a cascaded SRDF configuration" on page 138.

SRDF operations on the R1 side

Table 31 identifies the allowable SRDF actions when the R1 is the source of a TimeFinder/Snap or VP Snap session.

Table 31 Allowable SRDF operations when R1 is the source of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
createpair -restore	Y	Y	Y	Y	Y[2]	Y	Y[3]	Y	Y		
createpair -invalidate R1	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
createpair -invalidate R2	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
createpair -format	Y										
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
half_swap	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
swap -refresh R1	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
swap -refresh R2	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
establish	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
establish -full	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
restore -full	Y	Y	Y	Y	Y[2]	Y	Y[3]	Y	Y		

Table 31 Allowable SRDF operations when R1 is the source of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
update	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
failback	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
failover -restore	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]			Y[1]		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y	Y	Y	Y	Y	Y[3]	Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 31 Allowable SRDF operations when R1 is the source of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the devices are operating in async mode and there is a TimeFinder/Snap off of the R1 and either:
   a. The SRDF pair is an R1->R22 that will become an R2<-R21 and any of the following apply:
      i. What will become the R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or what will become the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of what will become the R21->R2 SRDF pair.
   b. The SRDF pair is an R1->R2 and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on what will become the R1 side.
2. Allowed for TimeFinder VP Snap. If not TimeFinder VP Snap, must use -force.
3. Only allowed for TimeFinder VP Snap.

Table 32 identifies the allowable SRDF actions when R1 is the target of a TimeFinder/Snap or VP Snap session.

Table 32 Allowable SRDF operations when R1 is the target of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -restore	Y					Y		Y	Y		
createpair -invalidate R1	Y					Y		Y	Y		
createpair -invalidate R2	Y					Y		Y	Y		
createpair -format	Y										
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y								Y		
half_swap	Y								Y		
swap -refresh R1	Y								Y		

Table 32 Allowable SRDF operations when R1 is the target of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
swap -refresh R2	Y								Y		
establish	Y					Y		Y	Y		
establish-full	Y					Y		Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y					Y		Y	Y		
restore -full	Y					Y		Y	Y		
update	Y					Y		Y	Y		
failback	Y					Y		Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y								Y		
failover -restore	Y								Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y					Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y					Y		Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y					Y		Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 32 Allowable SRDF operations when R1 is the target of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for TimeFinder/Snap R1 targets.

SRDF operations on the R2 side

Table 33 identifies the allowable SRDF actions when the R2 is the source of a TimeFinder/Snap or VP Snap copy session.

Table 33 Allowable SRDF operations when R2 is the source of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y[2]	Y[2]	Y[2]	Y[2,5]	Y[2]			Y[2]		
createpair -restore	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]			Y[2]		
createpair -invalidate R1	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]			Y[2]		
createpair -invalidate R2	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]			Y[2]		
createpair -format	Y										
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]		
half_movepair	Y	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]	Y[4]		

Table 33 Allowable SRDF operations when R2 is the source of a TimeFinder/Snap or VP Snap (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
swap	Y	Y	Y	Y	Y	Y		Y	Y		
half_swap	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R1	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R2	Y	Y	Y	Y	Y	Y		Y	Y		
establish	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]		Y[2]	Y[2]		
establish -full	Y	Y[2]	Y[2]	Y[2]	Y[2,5]	Y[2]		Y[2]	Y[2]		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[1]	Y[2]	Y[2]		
restore -full	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[1]	Y[2]	Y[2]		
update	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]		Y[2]	Y[2]		
failback	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]		Y[2]	Y[2]		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y	Y	Y	Y	Y	Y		Y	Y		
failover -restore	Y	Y	Y	Y	Y	Y		Y	Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y	Y	Y	Y	Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y		Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]		Y[2]	Y[2]		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 33 Allowable SRDF operations when R2 is the source of a TimeFinder/Snap or VP Snap (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]	Y[3]		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Only allowed if the required Enginuity levels or patches are detected. If TimeFinder VP Snap, requires Enginuity version 5876 Q42012 SR and higher.
2. Not allowed if the devices are operating in async mode and there is a TimeFinder/Snap off of the R2 and either:
   a. The SRDF pair will become an R21->R2 for which any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the R21->R2 SRDF pair.
      iii. If the R21->R2 pair will be RW on the SRDF link, the R21 must be pace-capable.
   b. The SRDF device pair is an R1->R2 and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side.
3. If the SRDF/A session is in the Transmit Idle state, you must issue the command with -symforce from the R1 side.
4. Not allowed if the devices are moving to a group operating in async mode and there is a TimeFinder/Snap off of the R2 and either:
   a. The SRDF pair is an R21->R2 and any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.
   b. The SRDF device pair is an R1->R2 and either:

i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.

5. If a TimeFinder VP Snap, you must use -force.

Table 34 identifies the allowable SRDF actions when the R2 is the target of a TimeFinder/Snap or VP Snap copy session.

Table 34 Allowable SRDF operations when R2 is the target of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y								Y		
createpair -restore	Y								Y		
createpair -invalidate R1	Y								Y		
createpair -invalidate R2	Y								Y		
createpair -format	Y										
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y					Y		Y	Y		
half_swap	Y					Y		Y	Y		
swap -refresh R1	Y					Y		Y	Y		
swap -refresh R2	Y					Y		Y	Y		
establish	Y					Y		Y	Y		
establish -full	Y					Y		Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y					Y		Y	Y		
restore -full	Y					Y		Y	Y		
update	Y					Y		Y	Y		
failback	Y					Y		Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y					Y		Y	Y		
failover -restore	Y					Y		Y	Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 34 Allowable SRDF operations when R2 is the target of a TimeFinder/Snap or VP Snap

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
merge	Y					Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y					Y		Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y					Y		Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

SRDF operations for TimeFinder/Clone sessions

This section lists the allowable SRDF operations for TimeFinder /Clone copy sessions on the R1 source and target and the R2 source and target.

SRDF operations on the R1 side

Table 35 identifies the allowable SRDF actions when the R1 is the source of a TimeFinder/Clone copy session.

Table 35 Allowable SRDF operations when R1 is the source of a TimeFinder/Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -invalidate R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -invalidate R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
half_swap	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
swap -refresh R1	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
swap -refresh R2	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
establish -full	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
restore -full	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
update	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
failback	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 35 Allowable SRDF operations when R1 is the source of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
failover -establish	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
failover -restore	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 35 Allowable SRDF operations when R1 is the source of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the devices are operating in async mode and there is a TimeFinder/Clone off of the R1 and either:
   a. The SRDF pair is an R22->R1which will become an R21<-R1 and any of the following apply:
      i. What will become the R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or what will become the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. The device that will become the R21 is not pace-capable.
      iii. SRDF/A device-level write pacing is not configured for autostart on the R2 side.
   b. The SRDF device pair is not an R22->R1 of a concurrent R2 setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R2 side.

Table 36 identifies the allowable SRDF actions when the R1 is the target of a TimeFinder/Clone copy session.

Table 36 Allowable SRDF operations when R1 is the target of a TimeFinder/Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y						Y			Y		Y	Y		
createpair -restore	Y						Y			Y		Y	Y		
createpair -invalidate R1	Y						Y			Y		Y	Y		
createpair -invalidate R2	Y						Y			Y		Y	Y		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 36 Allowable SRDF operations when R1 is the target of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
swap	Y												Y		
half_swap	Y												Y		
swap -refresh R1	Y												Y		
swap -refresh R2	Y												Y		
establish	Y					Y	Y			Y		Y	Y		
full establish	Y					Y	Y			Y		Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y						Y			Y		Y	Y		
restore -full	Y						Y			Y		Y	Y		
update	Y						Y			Y		Y	Y		
failback	Y						Y			Y		Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y												Y		
failover -restore	Y												Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y						Y			Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y						Y			Y		Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y					Y	Y			Y		Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 36 Allowable SRDF operations when R1 is the target of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for TimeFinder/Clone R1 targets.

SRDF operations on the R2 side

Table 37 identifies the allowable SRDF actions when the R2 is the source of a TimeFinder/Clone copy session.

Table 37 Allowable SRDF operations when R2 is the source of a TimeFinder/Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -restore	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -invalidate R1	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -invalidate R2	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y[4]	Y[4]	Y[4]	Y	Y	Y	Y[4]	Y[4]	Y	Y	Y	Y[4]		
half_movepair	Y	Y[4]	Y[4]	Y[4]	Y	Y	Y	Y[4]	Y[4]	Y	Y	Y	Y[4]		
swap	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
half_swap	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
establish	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
full establish	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y	Y[1,2]	Y	Y[2]		
restore -full	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y	Y[1,2]	Y	Y[2]		
update	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
failback	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
failover -restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 37 Allowable SRDF operations when R2 is the source of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y[3]	Y[3]	Y[3]	Y	Y	Y	Y[3]	Y[3]	Y	Y[3]	Y	Y[3]		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y[3]	Y[3]	Y[3]	Y	Y	Y	Y[3]	Y[3]	Y	Y[3]	Y	Y[3]		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Only allowed if required Enginuity levels or patches are detected and not a TimeFinder VP Snap.
2. Not allowed if the devices are in async mode and there is a TimeFinder/Clone off of the R2 either:
   a. The SRDF pair will become an R21->R2 for which any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side.
      iii. If the R21->R2 pair will be read/write (RW) on the SRDF link, the R21 must be pace-capable.
   b. The SRDF device pair is not an R21->R2 of a cascaded setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side.
3. If the SRDF/A session is in the Transmit Idle state, you must issue the command with -symforce from the R1 side.
4. Not allowed if the devices are moving to a group operating in async mode and there is a TimeFinder/Clone off of the R2 and either:
   a. The SRDF pair is an R21->R2 of a cascaded setup and any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.
   b. The SRDF device pair is not an R21->R2 of a cascaded setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.

Table 38 identifies the allowable SRDF actions when the R2 is the target of a TimeFinder/Clone copy session.

Table 38 Allowable SRDF operations when R2 is the target of a TimeFinder/Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y												Y		
createpair -restore	Y												Y		
createpair -invalidate R1	Y												Y		
createpair -invalidate R2	Y												Y		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y						Y			Y		Y	Y		
half_swap	Y						Y			Y		Y	Y		
swap -refresh R1	Y						Y			Y		Y	Y		
swap -refresh R2	Y						Y			Y		Y	Y		

Table 38 Allowable SRDF operations when R2 is the target of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
establish	Y						Y			Y		Y	Y		
establish -full	Y						Y			Y		Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y						Y			Y		Y	Y		
restore -full	Y						Y			Y		Y	Y		
update	Y						Y			Y		Y	Y		
failback	Y						Y			Y		Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y						Y			Y		Y	Y		
failover -restore	Y						Y			Y		Y	Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y						Y			Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y						Y			Y		Y	Y		
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y						Y			Y		Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 38 Allowable SRDF operations when R2 is the target of a TimeFinder/Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the TimeFinder/Clone R2 targets.

SRDF operations for Extent-level TimeFinder/Clone sessions

This section lists the allowable SRDF operations for Extent-level TimeFinder /Clone copy sessions on the R1 source and target and the R2 source and target.

SRDF operations on the R1 side

Table 39 identifies the allowable SRDF actions when the R1 is the source of an Extent-level TimeFinder/Clone copy session.

Table 39 Allowable SRDF operations when R1 is the source of an Extent-level Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -invalidate R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -invalidate R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
half_swap	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
swap -refresh R1	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
swap -refresh R2	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
establish -full	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
restore -full	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
update	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
failback	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 39 Allowable SRDF operations when R1 is the source of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
failover -establish	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
failover -restore	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y			Y[1]		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 39 Allowable SRDF operations when R1 is the source of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the devices are operating in async mode and there is a TimeFinder/Clone off of the R1 and either:
   a. The SRDF pair is an R22->R1 that will become an R21<-R1 and any of the following apply:
      i. The R22 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R1 Symmetrix array is running an Enginuity level lower than 5875.
      ii. The R21 that will result from the operation will not be pace-capable.
      iii. SRDF/A device-level write pacing is not configured for autostart on the R2 side.
   b. The SRDF device pair is not an R22->R1 of a concurrent R2 setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R2 side.

Table 40 identifies the allowable SRDF actions when the R1 is the target of an Extent-level TimeFinder/Clone copy session.

Table 40 Allowable SRDF operations when the R1 is the target of an Extent-level Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y														
createpair -restore	Y														
createpair -invalidate R1	Y														
createpair -invalidate R2	Y														
createpair -format	Y														
deletepair	Y														
half_deletepair	Y														
movepair	Y														
half_movepair	Y														
swap	Y														

Table 40 Allowable SRDF operations when the R1 is the target of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
half_swap	Y														
swap -refresh R1	Y														
swap -refresh R2	Y														
establish	Y					Y	Y			Y		Y	Y		
establish -full	Y					Y	Y			Y		Y	Y		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y														
restore -full	Y														
update	Y														
failback	Y														
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y														
failover -restore	Y														
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y														
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not _ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y														
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y					Y	Y			Y		Y	Y		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 40 Allowable SRDF operations when the R1 is the target of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the Extent-level TimeFinder/Clone R1 targets.

SRDF operations on the R2 side

Table 41 identifies the allowable SRDF actions when the R2 is the source of an Extent-level TimeFinder/Clone copy session.

Table 41 Allowable SRDF operations when R2 is the source of an Extent-level Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -restore	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -invalidate R1	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -invalidate R2	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y			Y[2]		
createpair -format	Y														
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y[4]	Y[4]	Y[4]	Y	Y	Y	Y[4]	Y[4]	Y	Y	Y	Y[4]		
half_movepair	Y	Y[4]	Y[4]	Y[4]	Y	Y	Y	Y[4]	Y[4]	Y	Y	Y	Y[4]		
swap	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
half_swap	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
swap -refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
establish	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
establish -full	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y	Y[1,2]	Y	Y[2]		
restore -full	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y	Y[1,2]	Y	Y[2]		
update	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
failback	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
failover -restore	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 41 Allowable SRDF operations when R2 is the source of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not _ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y		
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y[2]	Y[2]	Y[2]	Y	Y	Y	Y[2]	Y[2]	Y		Y	Y[2]		
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y[3]	Y[3]	Y[3]	Y	Y	Y	Y[3]	Y[3]	Y	Y[3]	Y	Y[3]		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y[3]	Y[3]	Y[3]	Y	Y	Y	Y[3]	Y[3]	Y	Y[3]	Y	Y[3]		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Only allowed if required Enginuity levels or patches are detected.
2. Not allowed if the devices are in async mode and there is a TimeFinder/Clone off of the R2 and either:
   a. The SRDF pair will become an R21->R2 for which any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side.
      iii. If the R21->R2 pair will be read/write (RW) on the SRDF link, the R21 must be pace-capable.
   b. The SRDF device pair is not an R21->R2 of a cascaded setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side.
3. If the SRDF/A session is in the Transmit Idle state, you must issue the command with -symforce from the R1 side.
4. Not allowed if the devices are moving to a group operating in async mode and there is a TimeFinder/Clone off of the R2 and either:
   a. The SRDF pair is an R21->R2 of a cascaded setup and any of the following apply:
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR, or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.
   b. The SRDF device pair is not an R21->R2 of a cascaded setup and either:
      i. The Enginuity level is lower than 5875 on either the R1 or R2 Symmetrix array.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 side of the new group.

Table 42 identifies the allowable SRDF actions when the R2 is the target of an Extent-level TimeFinder/Clone copy session.

Table 42 Allowable SRDF operations when the R2 is the target of an Extent-level Clone

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
createpair -establish	Y														
createpair -restore	Y														
createpair -invalidate R1	Y														
createpair -invalidate R2	Y														
createpair -format	Y														
deletepair	Y														
half_deletepair	Y														
movepair	Y														
half_movepair	Y														
swap	Y														
half_swap	Y														
swap -refresh R1	Y														

Table 42 Allowable SRDF operations when the R2 is the target of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
swap -refresh R2	Y														
establish	Y														
establish -full	Y														
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y														
restore -full	Y														
update	Y														
failback	Y														
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y														
failover -restore	Y														
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y														
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
refresh R2	Y														
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y														
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 42 Allowable SRDF operations when the R2 is the target of an Extent-level Clone (continued)

SRDF control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for Extent-level TimeFinder/Clone R2 targets.

SRDF set operations for TimeFinder/Snap sessions

This section lists the allowable SRDF set operations for TimeFinder /Snap copy sessions on the R1 source and target and the R2 source and target.

SRDF set operations on the R1 side

Table 43 identifies the allowable SRDF set actions when the R1 is the source of a TimeFinder/Snap copy session.

Table 43 Allowable SRDF set operations when R1 is the source of a TimeFinder/Snap

SRDF set operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

Table 44 identifies the allowable SRDF set actions when the R1 is the target for a TimeFinder/Snap copy session.

Table 44 Allowable SRDF set operations when R1 is the target of a TimeFinder/Snap

SRDF set operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

SRDF set operations on the R2 side

Table 45 identifies the allowable SRDF set actions when the R2 is the source of a TimeFinder/Snap copy session.

Table 45 Allowable SRDF set operations when R2 is the source of a TimeFinder/Snap

SRDF set operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1]	Y[1]		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. If the R2 is not an extent-based TimeFinder/Snap source device, then not allowed if any of the following are true:
   a. The RDF device pair is the R21->R2 of a cascaded setup and any of the following apply
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR or the R2 Symmetrix array is running an Enginuity level less than 5875.
      ii. The SRDF pair is read/write (RW) on the SRDF link and the R21 device is not pace-capable.
      iii. SRDF/A device-level write pacing is not configured for autostart on the R1 group of the R21 device.
   b. The SRDF pair is not the R21->R2 of a cascaded setup and either:
      i. The R1 or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 group.
   c. Not allowed if the Enginuity level is lower than 5671.

Table 46 identifies the allowable SRDF set when the R2 is the target of a TimeFinder/Snap copy session.

Table 46 Allowable SRDF set operations when R2 is the target of a TimeFinder/Snap

SRDF set operation:	No Session	Create in Progress	Created	Recreated	Copy on Write	Copied	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y										
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

SRDF set operations for TimeFinder/Clone sessions

This section lists the allowable SRDF set operations for TimeFinder /Clone copy sessions on the R1 and R2 source and the R1 and R2 target.

SRDF set operations on the R1 side

Table 47 identifies the allowable SRDF set actions when the R1 is the source of a TimeFinder/Clone copy session.

Table 47 Allowable SRDF set operations when R1 is the source of a TimeFinder/Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

Table 48 identifies the allowable SRDF set actions when the R1 is the target of a TimeFinder/Clone copy session.

Table 48 Allowable SRDF set operations when R1 is the target of a TimeFinder/Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y		Y	Y	Y	Y		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if TimeFinder/Clone pair was created with -copy.

SRDF set operations on the R2 side

Table 49 identifies the allowable SRDF set actions when the R2 is the source of a TimeFinder/Clone copy session.

Table 49 Allowable SRDF set operations when R2 is the source of a TimeFinder/Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y[1]	Y[1]	Y[1]	Y	Y	Y	Y[1]	Y[1]	Y		Y	Y[1]		
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. If the R2 is not an extent-based TimeFinder/Clone source device, then not allowed if any of the following are true:
   a. The RDF device pair is the R21->R2 of a cascaded setup and any of the following apply
      i. The R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR or the R2 Symmetrix array is running an Enginuity level less than 5875.
      ii. The SRDF pair is read/write (RW) on the SRDF link and the R21 device is not pace-capable.
      iii. SRDF/A device-level write pacing is not configured for autostart on the R1 group of the R21 device.
   b. The SRDF pair is not the R21->R2 of a cascaded setup and either:
      i. The R1 or the R2 Symmetrix array is running an Enginuity level lower than 5875.
      ii. SRDF/A device-level write pacing is not configured for autostart on the R1 group.
   c. Not allowed if the Enginuity level is lower than 5671.

Table 50 identifies the allowable SRDF set actions when the R2 is the target of a TimeFinder/Clone copy session.

Table 50 Allowable SRDF set operations when R2 is the target of a TimeFinder/Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y														
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

SRDF set operations for Extent-level TimeFinder/Clone sessions

This section lists the allowable SRDF set operations for Extent-level TimeFinder /Clone copy sessions on the R1 and R2 source and the R1 and R2 target.

SRDF set operations on the R1 side

Table 51 identifies the allowable SRDF set actions when the R1 is the source of an Extent-level TimeFinder/Clone copy session.

Table 51 Allowable SRDF set operations when R1 is the source of an Extent-level Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

Table 52 identifies the allowable SRDF set actions when the R1 is the target of an Extent-level TimeFinder/Clone copy session.

Table 52 Allowable SRDF set operations when R1 is the target of an Extent-level Clone

SRDF SET control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

SRDF set operations on the R2 side

Table 53 identifies the allowable SRDF set actions when the R2 is the source of an Extent-level TimeFinder/Clone copy session.

Table 53 Allowable SRDF set operations when the R2 is the source of an Extent-level Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

Table 54 identifies the allowable SRDF set actions when the R2 is the target of an Extent-level TimeFinder/Clone copy session.

Table 54 Allowable SRDF set operations when R2 is the target of an Extent-level Clone

SRDF set control operation:	No Session	Create in Progress	Created	Recreated	Precopy	Copy in Progress	Copied	Copy on Write	Copy on Access	Split	Restore in Progress	Restored	Term in Progress	Invalid	Failed
set mode async	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: There are no subscripts for the operations in this table.

APPENDIX B
SRDF Pair State Reference

This appendix describes the SRDF operations and their applicable pair states for:

- Standard R1/R2 configuration,
- Cascaded SRDF configuration,
- Concurrent SRDF configuration, and
- Composite group configuration.

Topics include:

- SRDF operations and applicable pair states 498
- Cascaded SRDF operations and applicable pair states 502
- Concurrent SRDF operations and applicable pair states 509
- Consistency group operations and applicable pair states 514

SRDF operations and applicable pair states

When SRDF control operations are initiated, the SRDF state of the device pair is first verified. If the device pair is not in a legal SRDF state to initiate the control operation, the action is blocked. However, by specifying the `-force` option, the control operation is performed, regardless of the pair state. If devices are running in SRDF/A mode, control operations for restore, update R1, and failback require the use of the -force option.

Examples:

The following initiates a failover on all SRDF pairs in the `prod` group that are in the Split state:

```
symrdf -g prod -force failover
```

The following initiates a failover on one SRDF pair, DEV001, in the `prod` group that is n the SyncInProg state:

```
symrdf -g prod -force failover DEV001
```

IMPORTANT

The `-force` option may place the SRDF pair into an undesirable state. After using this option, always check the pair state.

The first column in Table 55 lists the control operations you can invoke for the listed pair states. The allowable actions are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.9

Table 55 SRDF control operations and applicable pair states (page 1 of 3)

Control operation:	R1 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
deletepair			Y[28,36,37]	Y[28,36,37]	Y[28,36,37]							
half_deletepair			Y[28,36,37]	Y[28,36,37]	Y[28,36]	Y[28,36,37]	Y[28,36,37]					
movepair			Y	Y	Y							
half_movepair			Y	Y	Y	Y	Y					
swap			Y[43]	Y[6,43]	Y[43]			Y[43]				
swap -refresh R1			Y[29,43]	Y[6,29,43]	Y[29,43]			Y[29,43]				
swap -refresh R2			Y[29,43]	Y[6,29,43]								
half_swap			Y[43]	Y[43]	Y[43]	Y[43]	Y[43]					

Table 55 SRDF control operations and applicable pair states (page 2 of 3)

Control operation:	R1 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
establish			Y[26]	Y[26]	Y[5,26]			Y[5,26]		Y[12]		
establish -full			Y[26]	Y[26]	Y[5,26]			Y[5,26]		Y[12, 17]		
split	Y[15,19, 28,36,37]	Y		Y[19,28, 36,37]				Y[19,28, 36,37]			Y[19,28, 36,37]	
restore			Y[29]	Y[3,29]	Y[5,29]			Y[5,29]		Y[13]		
restore -full			Y[29]	Y[3,29]	Y[5,29]			Y[5,29]		Y[13,14]		
update				Y[1,28, 29]	Y[28,29]			Y[28,29]				
failback			Y[26,28, 29]	Y[1,26, 28,29]	Y[26,28, 29]	Y[8,18, 26]	Y[8,18, 26]	Y[26,28, 29]	Y[26,28, 29]			
failover	Y[15,19, 30,41,42]	Y[19,30, 41,42]	Y[18,19, 30,41, 42]	Y[19,30, 41,42]		Y[9,19, 41,42]	Y[9,19, 41,42]	Y[19,30, 41,42]	Y[15,19, 30,41, 42]	Y[18,19 30, 41, 42]	Y[19,30, 41,42]	Y[19,30, 41,42]
failover -establish	Y[15,17, 29,30 43]	Y[29, 30, 43]	Y[18,19, 30,43]	Y[29,30, 43]	Y[29,30, 43]			Y[29,30, 43]	Y[15,29, 30,43]	Y[18,29, 30,43]		
failover -restore	Y[15,26, 31, 43]	Y[26,31, 43]	Y[18,26, 31,43]	Y[26,31, 43]	Y[18,26, 31, 43]					Y[18,26, 31,43]	Y[26,31 43]	
invalidate R1				Y[2,29]								
invalidate R2				Y[26]								
merge				Y[3,33, 34]	Y[18,33, 34]							
msc_cleanup			Y	Y	Y	Y[9]	Y[9]					
not_ready R1	Y[6,21]	Y[6,21]	Y[6,21]	Y[6,21]	Y[21]	Y[6,8, 21]	Y[6,8, 21]	Y[21]	Y[21]	Y[6,21]	Y[6,21]	Y[6,8,21]
not_ready R2	Y[10,17, 21]	Y[10,21]	Y[10,21]	Y[21]	Y[10,21]	Y[7,9, 21]	Y[7,9, 21]	Y[10,21]	Y[10, 21]	Y[10,17, 21]		
ready R1	Y[21]	Y[21]	Y[21]	Y[21]	Y[21]	Y[8,21]	Y[8,21]	Y[21]	Y[21]	Y[21]	Y[21]	Y[21]
ready R2	Y[17,21]	Y[21]	Y[21]	Y[21]	Y[21]	Y[9,21]	Y[9,21]	Y[21]	Y[21]	Y[17,21]		
refresh R1				Y[1,20, 39]	Y[18,39]							
refresh R2				Y[1,18, 26]								
resume				Y								

Table 55 SRDF control operations and applicable pair states (page 3 of 3)

Control operation:	R1 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
suspend	Y[15,19,28,36,37,38]	Y	Y[16,18,19,28,36,37,38]	Y[7,19,28,36,37,38]	Y[5,19,28,36,37,38]	Y[19,28,36,37,38]	Y[19,28,36,37,38]	Y[19,28,36,37,38]	Y[15,19,28,36,37,38]	Y[18,19,28,36,37,38]	Y[19,20,28,36,37,38]	Y[19,20,28,36,37,38]
disable	Y[41]	Y[41]	Y[41]	Y[41]	Y[41]	Y[18,41]	Y[18,41]	Y[41]	Y[41]	Y[41]	Y[41]	Y[23]
enable	Y[24]		Y[24]	Y[24]	Y[24]			Y[24]	Y[24]	Y[24]	Y	Y[23]
rw_disable R2	Y[17]	Y	Y[11]	Y	Y[11]	Y[7,9]	Y[7,9]	Y[11]	Y[11]	Y[2,17]		
rw_enable R1	Y[1]	Y[1]		Y[1]	Y[18]	Y[1,8]	Y[1,8]			Y[1]	Y[1]	Y[1,8]
rw_enable R2				Y[7]		Y[7,9]	Y[7,9]					
write_disable R1	Y[18]	Y[18]	Y[18]	Y[18]		Y[4,8]	Y[4,8]			Y[18]	Y[18]	Y[4,8]
write_disable R2			Y[18]	Y[18]	Y[18]	Y[9,10]	Y[9,10]	Y[18]	Y[18]	Y[18]		Y[9,10]
activate -rdfa_dse	Y[22]										Y	Y[23]
deactivate -rdfa_dse	Y[22]										Y	Y[23]
activate -rdfa_devpace	Y[22]										Y	Y[23]
deactivate -rdfa_devpace	Y[22]										Y	Y[23,25]
activate -rdfa_pace	Y[22]										Y	Y[23]
deactivate -rdfa_pace	Y[22]										Y	Y[23,25]
activate -rdfa_wpace	Y[22]										Y	Y[23]
deactivate -rdfa_wpace	Y[22]										Y	Y[23]
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y			Y	Y		Y	
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y			Y	Y		Y	

1. SA is Write Disabled or is Not Ready on the source side.
2. SA is Write Disabled, or is Not Ready on the source side, or must use -nowd.
3. SA is Write Disabled, or is Not Ready on the source side, or must use -force.
4. SA is Ready on the source side.
5. Source is not visible to any host.
6. Write Disabled on the source.
7. SA or RA is Write Disabled or is Not Ready on the target side.
8. Host application running while connected to the source.
9. Host application running while connected to the target.
10. RA is Ready on the target side.
11. RA is Write Disabled on the target side.
12. Source and target are Not Ready but the SRDF link is Ready and there are no local or remote invalid tracks on the source or the target.
13. Source and target are Not Ready but the SRDF link is Ready and there are no remote invalid tracks on the source side.
14. Source and target are Not Ready but the SRDF link is Ready and there are no local or remote invalid tracks on the source side.
15. Can use -symforce.
16. Write Disabled on the SRDF link.
17. Not allowed when SRDF/A is active.
18. Must use -force.
19. If enabled for SRDF consistency protection, must use -force.
20. Must use -immediate.
21. Not allowed on a diskless device.
22. SRDF/A must be active.
23. Source must be reachable.
24. Must be in async mode.
25. Only allowed on the R1 side and must use -symforce.
26. No local invalid tracks on the source side.
27. No remote invalid tracks on the source side.
28. If remote invalid tracks are on the source side, must use -force.
29. No local invalid tracks on the target side.
30. If remote invalid tracks are on the source side, must use -symforce.
31. If remote invalid tracks are on the target side, must use -force.
32. No remote invalid tracks on the target side.
33. Source device is Read Write Enabled and there are no local and remote invalid tracks on the target side.
34. Target device is Read Write Enabled and there are no local and remote invalid tracks on the target side.
35. Source and target devices are Read Write Enabled.
36. If there are local invalid tracks on the source side, must use -symforce if the source is not an R11 or R21.
37. If there are local invalid tracks on the target side, must use -symforce if the target is not an R11 or R21.
38. Write Disabled on the SRDF link and must use -force.
39. No local invalid tracks on the target side and no remote invalid tracks on the source side and must use -force.
40. If the SRDF device pair is operating in adaptive copy mode, must use -symforce.
41. If enabled for CG SRDF consistency protection, must use -force.
42. If local invalid tracks are on the target side, must use –symforce.
43. Not allowed if enabled for SRDF consistency protection.

Cascaded SRDF operations and applicable pair states

Table 56 lists the allowable control operations for the R1->R21 pair given the pair states for the R21->R2 pair. The allowable control actions are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 56 R1->R21 cascaded SRDF control operations and applicable pair states (page 1 of 3)

R1 ->R21 control operation:	R21 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
createpair -establish	Y[1,3,4,16,17,21]	Y[1,3,21]	Y[1,12,21]	Y[1,12,21]	Y[1,12,20]	Y[1,12,21]					Y[1,3,4,16,17]	Y[1,3,4,16,17]
createpair -restore	Y[1,5,16,17]	Y[1]	Y[1,12,21]	Y[1,12,21]	Y[1,12,20]	Y[1,6,12]					Y[1,6,16,17]	Y[1,5,16,17]
createpair -invalidate R1	Y[1,5,16]	Y[1,6]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1]	Y[1]		Y[1,16]	Y[1,5,16]
createpair -invalidate R2	Y[1,6,16]	Y[1,6]	Y[1,6]	Y[1,6]	Y[1,6]	Y[1,6]		Y[1,6]	Y[1,6]		Y[1,16]	Y[1,16]
deletepair	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,12]	Y[1,12]	Y[1,12]	Y[1,13]	Y[1,12]	Y[1,12]
half_deletepair	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,12]	Y[1,12]	Y[1,12]	Y[1,13]	Y[1,12]	Y[1,12]
movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
swap	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1,12]	Y[1,12]		Y[1,12]	Y[1,12]
half_swap	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,12]	Y[1,12]	Y[1,12]		Y[1,12]	Y[1,12]
swap -refresh R1	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1,12]	Y[1,12]		Y[1,12]	Y[1,12]
swap -refresh R2			Y[1]	Y[1]	Y[1,20]	Y[1]						
establish	Y[4,17]	Y[2]	Y[12]	Y[12]	Y[12,20]	Y[12]					Y[2,4,17]	Y[2,3,4,17]
establish -full	Y[3,4,17]	Y[3]	Y[12]	Y[12]	Y[12,20]	Y[12]					Y[2,3,4,17]	Y[2,3,4,17]
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y[5,17]	Y	Y[12]	Y[12]	Y[12,20]	Y[6,12]					Y[17]	Y[5,6,17]
restore -full	Y[5,17]	Y		Y[12]	Y[12,20]	Y[6,12]					Y[17]	Y[5,6,17]
update	Y[5,17]	Y	Y[12]	Y[12]	Y[12]	Y[12]		Y	Y		Y[17]	Y[5,17]
failback	Y[5,17]	Y	Y[12]	Y[12]	Y[12,20]	Y[6,12]					Y	Y[5,6,17]
failover	Y	Y	Y	Y	Y	Y	Y			Y	Y	Y

Table 56 R1->R21 cascaded SRDF control operations and applicable pair states (page 2 of 3)

R1 ->R21 control operation:	R21 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
failover -establish	Y[1,12]	Y[1,12]	Y[1,12]	Y[1,12]	Y[1,12,20]	Y[1,12]					Y[1,12]	Y[1,12]
failover -restore	Y[1,8,11,12]	Y[1,11,12]	Y[1,12]	Y[1,12]	Y[1,12,20]	Y[1,12]	Y[1,11,12]			Y[11,12]	Y[1,8,11,12]	Y[1,8,11,12]
invalidate -R1	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
merge	Y[14]	Y[14]	Y[14]	Y[14]	Y[14]	Y[14]		Y[14]	Y[14]		Y[14]	Y[14]
msc_cleanup												
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
refresh R2			Y	Y	Y	Y		Y	Y			
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y[4,5,17]	Y[5]	Y[12]	Y[12]	Y[12,20]	Y[12]					Y[2,4,17]	Y[2,3,4,5,17]
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 56 R1->R21 cascaded SRDF control operations and applicable pair states (page 3 of 3)

R1 ->R21 control operation:	R21 ->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. If the other pair (the one not being controlled) is enabled for SRDF consistency protection, must use -force. This operation can change the composite group type, causing SRDF consistency monitoring to stop.
2. If the other pair (not being controlled) is enabled for SRDF consistency protection, must use -force.
3. Must use -force.
4. Not allowed if operation results in R1->R21<-R2 data resychronization.
5. Not allowed if R21 is diskless and operation will result in R1<-R21->R2 data resynchronization.
6. If tracks are owed to R21 while R21->R2 is in the Transmit Idle state, data resynchronization between R1->R21 cannot complete.
7. Not allowed if operation results in local invalid tracks on the R21 device.
8. Not allowed if R2 owes tracks to R21.
9. Not allowed if operation results in data flowing from R2->R21.
10. Not allowed if operation creates a concurrent R22 device on a Symmetrix array running on an Enginuity level lower than 5773.150.
11. Must use -remote.
12. Not allowed if R21 is diskless.
13. Not allowed if R1 is diskless and the SRDF link of the other pair is RW.
14. Not allowed if R21 is diskless and both mirrors of R21 have invalid tracks.
15. Must use -force. The state of the other pair changes to Suspended.
16. Not allowed if SRDF/A group-level write pacing or SRDF/A device-level write pacing is active and supported on the R1 mirror of what will become the R21 and the R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR.
17. If the pair being controlled is the R1->R21 pair and is operating in adaptive copy mode and the R1 mirror of the R21 has either SRDF/A group-level or SRDF/A device-level write pacing activated and supported, must use -force.
18. If the pair being controlled is (or will become) the R21->R2 pair and is operating in asynchronous mode with SRDF/A device-level or group-level write pacing configured for autostart on the R1 mirror of the R21, and the R1->R21 pair is operating in adaptive copy mode and is read/write (RW) on the SRDF link, must use -force.
19. If the R1->R21 pair is operating in adaptive copy mode and is read/write (RW) on the SRDF link, must use –force.
20. The R21 is not visible to any host.
21. Not allowed when what will become the R21->R2 is in Synchronous mode.

Table 57 lists the allowable control operations for the R21->R2 pair given the SRDF pair states for the R1->R21 pair. These allowable actions are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 57 R21->R2 cascaded SRDF control operations and applicable pair states (page 1 of 3)

	R1 ->R21 pair state:											
R21->R2 control operation:	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
createpair -establish	Y[1,18]	Y[1,18]	Y[1,5]	Y[1,5]	Y[1,5]	Y[1,5]		Y[1]	Y[1]		Y[1]	Y[1]
createpair -restore			Y[1,6]	Y[1]	Y[1,6]	Y[1,6]		Y[1]	Y[1]			
createpair -invalidate R1	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,6]	Y[1,6]		Y[1]	Y[1]		Y[1]	Y[1]
createpair -invalidate R2	Y[1]	Y[1]	Y[1,5]	Y[1,5]	Y[1,5]	Y[1,5]		Y[1]	Y[1]		Y[1,6]	Y[1]

Table 57 R21->R2 cascaded SRDF control operations and applicable pair states (page 2 of 3)

R21->R2 control operation:	R1 ->R21 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
deletepair	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,12]	Y[1,12]	Y[1,12]	Y[1]	Y[1,12]	Y[1,12]
half_deletepair	Y[1,12]	Y[1,12]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,12]	Y[1,12]	Y[1,12]	Y[1]	Y[1,12]	Y[1,12]
movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
swap	Y[1,10,12]	Y[1,10,12]	Y[1,10]	Y[1,10]	Y[1,10]	Y[1,2,10]		Y[1,10,12]	Y[1,2,10]		Y[1,10,12]	Y[1,10,12]
half_swap	Y[1,10,12]	Y[1,10,12]	Y[1,10]	Y[1,10]	Y[1,10]	Y[1,10]	Y[1,10,12]	Y[1,10,12]	Y[1,10,12]		Y[1,10,12]	Y[1,10,12]
swap -refresh R1	Y[1,10,12]	Y[1,10,12]	Y[1,10]	Y[1,10]	Y[1,10]	Y[1,10]		Y[1,10,12]	Y[1,10,12]		Y[1,10,12]	Y[1,10,12]
swap -refresh R2	Y[1,10,12]	Y[1,10,12]	Y[1,10]	Y[1,10]	Y[1,10]	Y[1,10]		Y[1,10,12]	Y[1,10,12]		Y[1,10,12]	Y[1,10,12]
establish	Y[18]	Y[18]	Y[5]	Y[5]	Y[5]	Y[5]		Y	Y		Y	Y
full establish	Y[18]	Y[18]	Y[5]	Y[5]	Y[5]	Y[5]		Y	Y		Y	
split	Y[12]	Y[12]	Y	Y	Y	Y	Y[12]	Y[12]	Y[12]	Y[13]	Y	Y
restore			Y	Y	Y	Y		Y	Y			
full restore			Y	Y	Y	Y		Y	Y			
update			Y	Y	Y	Y		Y	Y			
failback			Y	Y	Y	Y[12]		Y	Y			
failover			Y	Y	Y	Y	Y	Y[12]	Y[12]	Y[13]		
failover -establish			Y[1,10,12]	Y[1,10,12]	Y[1,10,12,20]	Y[1,10]						
failover -restore			Y[1,5,10,12]	Y[1,10,12]	Y[1,10,12,20]							
invalidate -R1	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
merge	Y[5,7]	Y[5,7]	Y[5]	Y[5]	Y[5,7]	Y[5,7]		Y[5,7]	Y[5,7]		Y[5,7]	Y[5,7]
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 57 R21->R2 cascaded SRDF control operations and applicable pair states (page 3 of 3)

R21->R2 control operation:	R1 ->R21 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
refresh R1			Y	Y	Y[20]	Y						
refresh R2	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
suspend	Y[12]	Y[12]	Y	Y	Y	Y	Y[12]	Y[12]	Y[12]	Y[13]	Y	Y
resume	Y[9,18]	Y[9,18]	Y[5]	Y[5]	Y[5]	Y[5]		Y	Y		Y[9]	Y[6,9]
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y[19]	Y[19]	Y	Y	Y	Y		Y[19]	Y[19]		Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y[19]	Y[19]	Y	Y	Y	Y		Y[19]	Y[19]		Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y[19]	Y[19]	Y	Y	Y	Y		Y[19]	Y[19]		Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: Table 57 uses the same keys as Table 56 on page 502.

Table 58 lists the allowable set operations for the R1->R21 pair given the pair states for the R21->R2 pair. The allowable control actions are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 58 R1->R21 cascaded SRDF set operations and applicable pair states

R1->R21 set operation:	R21->R2 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
set mode async[2]	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y[1]	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[1]	Y[1]
set mode acp_wp	Y[1]	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[1]	Y[1]

1. You must use -force if SRDF/A device-level and/or group-level write pacing is activated and supported for the SRDF/A session that includes the R21->R2 RDF device pair, and the R1->R21 SRDF device pair (that is being controlled) is read/write (RW) on the SRDF link.
2. Async mode is not supported on both sides of an R21. It is only supported on one side or the other.

Table 59 lists the allowable set operations for the R21->R2 pair given the pair states for the R1->R21 pair. The allowable control actions are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 59 R21->R2 Cascaded RDF Set Operations and Applicable Pair States

R21->R2 set operation:	R1->R21 pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
set mode async	Y[1]	Y[1]	Y	Y	Y	Y	Y[2]	Y[1]	Y[1]	Y[2]	Y	Y
set mode sync	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_disk	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
set mode acp_wp	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Must use -force if all of these conditions apply:
   d. SRDF/A device-level and/or group-level write pacing is configured for autostart for the SRDF/A session that includes the R21->R2 SRDF device pair.
   e. The R21->R2 SRDF device pair (that is being controlled) is read/write (RW) on the SRDF link.
   f. The R1->R21 SRDF device pair (that is not being controlled) is operating in adaptive copy mode and is read/write (RW) on the SRDF link.
2. Not allowed if SRDF/A device-level and/or group-level write pacing is configured for autostart for the SRDF/A session that includes the R21->R2 device pair.

Concurrent SRDF operations and applicable pair states

This section provides the concurrent SRDF control operations and their applicable pair states for concurrent R1 (R11) and concurrent R2 (R22).

Concurrent R1

In a concurrent R1 relationship, there are two separate links, or legs, sending data from one R1 device to two separate R2 mirrors. You can perform a control operation on one of these legs only if the other leg is in a certain pair state.

Note: If a concurrent R1 device is made RW (read write) from either of the SRDF relationships, it is also seen as RW from the other relationship. The commands to make a concurrent R1 device RW are: `rw_enable R2`, `split`, and `failover`.

Table 60 lists the allowable control operations for the first leg of the concurrent R1 pair (the one being controlled by an SRDF action) given the pair state of the second leg (the one not being controlled). The allowable operations are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 60 SRDF control operations and applicable states for concurrent R1 pairs (page 1 of 3)

Control operation of 1st leg of concurrent SRDF R1 pair:	Pair state of 2nd leg of concurrent SRDF R1 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
createpair -establish	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
createpair -restore	Y[1,5,6]	Y[1,6]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,6]			Y[1,5]	Y[1,5,6]	Y[1,5,6]
createpair -invalidate R1	Y[1,5]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,5]	Y[1]	Y[1,5]	Y[1,5]
createpair -invalidate R2	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
deletepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_deletepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
swap	Y[1,12,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,12]	Y[1,12]
half_swap	Y[1,12,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]		Y[1,12]	Y[1]
swap -refresh R1	Y[1,5,12,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,5,17]	Y[1,17]	Y[1,5,12]	Y[1,5,12]
swap -refresh R2	Y[1,12,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,17]	Y[1,12]	Y[1,12]
establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
full establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 60 SRDF control operations and applicable states for concurrent R1 pairs (page 2 of 3)

Control operation of 1st leg of concurrent SRDF R1 pair:	Pair state of 2nd leg of concurrent SRDF R1 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
split	Y	Y	Y	Y	Y[2]	Y	Y	Y[3]		Y	Y	Y
restore	Y[5,6]	Y[6]	Y	Y	Y	Y	Y[6]			Y[6]	Y[5,6]	Y[5,6]
full restore	Y[5,6]	Y[6]	Y	Y	Y	Y	Y[6]			Y[6]	Y[5,6]	Y[5,6]
update	Y[5,6]	Y[6]	Y	Y	Y	Y	Y			Y[6]	Y[5,6]	Y[5,6]
failback	Y[5]	Y	Y[5]	Y[5]	Y[5]	Y[5]	Y[5]	Y	Y[5]	Y[6]	Y[5]	Y[5]
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y[1,12,13]	Y[1]	Y[1,8]	Y[1,8]	Y[1,8,15]	Y[1,8]	Y[1,8]				Y[1,12,13]	Y[1,12,13]
failover -restore	Y[1,11,12,13]	Y[1]	Y[1,8]	Y[1,8]	Y[1,8,15]	Y[1,8]	Y[1,8]				Y[1,12,13]	Y[1,4,11,12,13]
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y[5]	Y	Y	Y	Y	Y	Y	Y	Y[5]		Y[5]	Y[5]
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y[5]	Y	Y	Y	Y	Y	Y	Y	Y[5]		Y[5]	Y[5]
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y[5,7]	Y[7]	Y	Y	Y	Y	Y[6]	Y	Y[5]	Y[5,6]	Y[5,7]	Y[5,7]
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 60 SRDF control operations and applicable states for concurrent R1 pairs (page 3 of 3)

Control operation of 1st leg of concurrent SRDF R1 pair:	Pair state of 2nd leg of concurrent SRDF R1 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. If the other pair (the one not being controlled) is enabled for SRDF consistency protection, must use -force. This operation can change the composite group type, causing SRDF consistency monitoring to stop.
2. Must use -force. The state of the other pair changes to Suspended.
3. Changes the state of the other pair to Split.
4. If device will become an R21 and the other pair is in Transmit Idle pair state, data synchronization between R1->R21 cannot complete.
5. Not allowed if R2->R11<-R2 data resynchronization will result.
6. Must use -remote.
7. Only allowed if data flows from R2 to R11 and -remote is used.
8. Not allowed if R11 is diskless.
9. Not allowed if a diskless device and will become an R21 device.
10. Not allowed if tracks are owed to R21 from concurrent R2.
11. Not allowed if a diskless device and will become an R21 device and results in R1<-R21->R2 data resynchronization.
12. Not allowed if SRDF/A group-level write pacing or SRDF/A device-level write pacing is active and supported on the other R1 mirror what will become the R21 and the R21 Symmetrix array is running an Enginuity level lower than 5876 Q42012 SR.
13. If the pair being controlled is operating in adaptive copy mode, must use -force if the R1 mirror of what will become the R21 has SRDF/A group-level and/or device-level write pacing activated and supported.
14. If the pair being controlled is operating in asynchronous mode, with SRDF/A group-level and/or device-level write pacing enabled for autostart on what will be the R1 mirror of the resulting R21, must use -force if the other pair (that is not being controlled) is operating in adaptive copy mode.
15. The R11 is not visible to any host.
16. The other pair's (the one not being controlled) R1 is not visible to any host.
17. Not allowed when what will become the R21->R2 is in Synchronous mode.

Concurrent R2

Concurrent R2 devices are intended for SRDF/Star configurations. In a concurrent R2 configuration, an R2 device has two remote mirrors, only one of which can be active (read/write) at a given time. For detailed information, refer to the *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*.

Table 61 lists the allowable control operations for the first leg (the one being controlled by an SRDF action) of the concurrent R2 pair given the pair state of the second leg (the one not being controlled). The allowable operations are noted by Ys.

The Partitoned1 pair state indicates that the remote Symmetrix is in the SYMAPI database and was discovered. The Partitioned2 pair state indicates the remote Symmetrix is not in the SYMAPI database and was not discovered, or was removed from this database.

Table 61 SRDF control operations and applicable states for concurrent R2 pairs (page 1 of 3)

Control operation of 1st leg of concurrent SRDF R2 pair:	Pair state of 2nd leg of concurrent SRDF R2 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
createpair -establish			Y[1,2]	Y[1]	Y[1,2]	Y[1]						
createpair -restore			Y[1,2]	Y[1]	Y[1,2]	Y[1]						
createpair -invalidate R1	Y[1]	Y[1]	Y[1,2]	Y[1]	Y[1,2]	Y[1]		Y[1]	Y[1]		Y[1]	Y[1]
createpair -invalidate R2	Y[1]	Y[1]	Y[1,2]	Y[1]	Y[1,2]	Y[1]		Y[1]	Y[1]		Y[1]	Y[1]

Table 61 SRDF control operations and applicable states for concurrent R2 pairs (page 2 of 3)

Control operation of 1st leg of concurrent SRDF R2 pair:	Pair state of 2nd leg of concurrent SRDF R2 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
deletepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_deletepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
half_movepair	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]
swap	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1]	Y[1]		Y[1]	Y[1]
half_swap	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1]	Y[1]		Y[1]	Y[1]
swap -refresh R1	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]		Y[1]	Y[1]		Y[1]	Y[1]
swap -refresh R2				Y[1]		Y[1]						
establish			Y[2]	Y	Y[2]	Y						
establish -full			Y[2]	Y	Y[2]	Y						
split			Y	Y[3]	Y	Y		Y	Y	Y		
restore			Y[2]	Y	Y[2]	Y						
restore -full			Y[2]	Y	Y[2]	Y						
update			Y	Y	Y	Y						
failback			Y[2]	Y	Y[2]	Y						
failover			Y	Y[3]	Y	Y	Y			Y		
failover -establish			Y	Y[3]	Y[16]	Y						
failover -restore			Y[1]	Y[1,3]	Y[1,16]	Y[1]		Y[1,14]	Y[1,14]			
invalidate -R1	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
merge			Y	Y	Y	Y						
msc_cleanup			Y	Y	Y	Y	Y					
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y[2]	Y	Y[2]	Y	Y	Y[2]	Y[2]	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y		Y	Y		Y	Y
refresh R2				Y		Y						
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume			Y[2]	Y	Y[2]	Y						

Table 61 SRDF control operations and applicable states for concurrent R2 pairs (page 3 of 3)

Control operation of 1st leg of concurrent SRDF R2 pair:	Pair state of 2nd leg of concurrent SRDF R2 pair:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
rw_disable R2	Y	Y	Y[2]	Y	Y[2]	Y		Y[2]	Y[2]		Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y		Y	Y	Y	Y	Y
rw_enable R2			Y	Y[3]	Y	Y	Y	Y	Y	Y		
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y[2]	Y	Y[2]	Y		Y[2]	Y[2]		Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Note: Table 61 uses the same keys as Table 60 on page 509.

Consistency group operations and applicable pair states

This section provides the consistency group (SRDF/CG) control operations and the applicable pair states for devices within a consistency group.

Table 62 SRDF control operations and applicable pair states for devices in an SRDF/CG

Control operation:	Pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
enable	Y	Y	Y	Y	Y			Y	Y	Y	Y	
disable	Y	Y	Y	Y	Y	Y[2]	Y[2]	Y	Y	Y	Y	Y[2]

Table 62 SRDF control operations and applicable pair states for devices in an SRDF/CG

Control operation:	Pair state:											
	SyncInProg	Synchronized	Split	Suspended	Failed Over	Partitioned1	Partitioned2	R1 Updated	R1 UpdInProg	Invalid	Consistent	TransmitIdle
modify -add	Y[1]	Y									Y	
modify -remove	Y[1]	Y									Y	
modify -recover	Y	Y		Y		Y[2]	Y[2]				Y	

1. There are no local invalid tracks on the source side and no remote invalid tracks on the remote side.
2. Must use -force.

APPENDIX C
Rcopy State Rules Reference

This appendix lists the SRDF control operations that are allowed for devices in various Open Replicator copy (Rcopy) pair states. Topics include:

- Rcopy Session on the R1 side..... 518
- Rcopy Session on the R2 side..... 522

Rcopy Session on the R1 side

This section lists the allowable SRDF operations with their applicable Rcopy states when there is an Rcopy session on the R1.

R1 is part of an Rcopy PUSH

Table 63 identifies the allowable SRDF actions with their applicable Rcopy states when there is an Rcopy PUSH session on the R1.

Table 63 Allowable SRDF operations with Rcopy PUSH session on the R1

Rcopy State: / SRDF Control Operation:	None	Create in Progress	Created	Copy in Progress	Copy on Write	Copied	Recreate in Progress	Recreated	Terminate in Progress	Failed	Invalid	Verify In Progress	Restored	Restored In Prog	Precopy	Sync In Prog	Synchronised	Stopped
createpair -establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[1,6]	Y	Y	Y	Y
createpair -restore	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
createpair -invalidate R1	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
createpair -invalidate R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[1,6]	Y	Y	Y	Y	Y
createpair -format	Y																	
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y																	
half_swap	Y																	
swap -refresh R1	Y																	
swap -refresh R2	Y																	
establish	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[4]	Y[1,6]	Y	Y	Y	Y
establish -full	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[4]	Y[1,6]	Y	Y	Y	Y
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
restore -full	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
update R1	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
failback	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y																	
failover -restore	Y																	

Table 63 Allowable SRDF operations with Rcopy PUSH session on the R1 (continued)

Rcopy State: / SRDF Control Operation:	None	Create in Progress	Created	Copy in Progress	Copy on Write	Copied	Recreate in Progress	Recreated	Terminate in Progress	Failed	Invalid	Verify In Progress	Restored	Restored In Prog	Precopy	Sync In Prog	Synchronised	Stopped
invalidate -R1	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[6]	Y[1,6]	Y	Y	Y	Y
merge	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y		Y[1]			Y[1]		Y[1]					Y[1,6]					
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[6]	Y[1,6]	Y	Y	Y	Y
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y[2]	Y[5]	Y[2]	Y[2]	Y[5]	Y[2]	Y[5]	Y[2]	Y[2]	Y[2]	Y[2]	Y[5,6]	Y[2,6]	Y[2]	Y[2]	Y[2]	Y[2]
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the R1 Symmetrix array is running an Enginuity level lower than 5874.
2. Not allowed if the R2 owes data to the R1.
3. If the R2 owes data to the R1, not allowed if the R1 Symmetrix array is running an Enginuity level lower than 5874, or if donor update is specified.
4. Not allowed if the Rcopy session has front end zero detect.
5. If R2 owes data to the R1, not allowed if the R1 Symmetrix is running an Enginuity level less than 5874 or if donor update specified.
6. Not allowed if the Rcopy session has front end zero detect.

R1 is part of an Rcopy PULL

Table 64 identifies the allowable SRDF actions with their applicable Rcopy states when there is an Rcopy PULL session on the R1.

Table 64 Allowable SRDF operations with Rcopy PULL session on the R1

Rcopy State: / SRDF Control Operations:	None	Create in Progress	Created	Copy In Progress	Copy on Access	Copied	Terminate In Progress	Failed	Invalid	Verify In Progress	Sync In Prog	Synchronised	Stopped	Failback
createpair -establish	Y													
createpair -restore	Y													
createpair -invalidate R1	Y													
createpair -invalidate R2	Y													
createpair -format	Y													
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y													
half_swap	Y													
swap -refresh R1	Y													
swap -refresh R2	Y													
establish	Y	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]
establish-full	Y	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]	Y[1,3]
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y													
restore -full	Y													
update	Y													
failback	Y													
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 64 Allowable SRDF operations with Rcopy PULL session on the R1 (continued)

Rcopy State: / SRDF Control Operations:	None	Create in Progress	Created	Copy In Progress	Copy on Access	Copied	Terminate In Progress	Failed	Invalid	Verify In Progress	Sync In Prog	Synchronised	Stopped	Failback
failover -establish	Y													
failover -restore	Y													
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]	Y[2]
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y													
refresh R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y	Y[1,2,3]	Y[1,3]	Y[1,2,3]	Y[1,2,3]	Y[1,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]	Y[1,2,3]
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 64 Allowable SRDF operations with Rcopy PULL session on the R1 (continued)

Rcopy State: / SRDF Control Operations:	None	Create in Progress	Created	Copy In Progress	Copy on Access	Copied	Terminate In Progress	Failed	Invalid	Verify In Progress	Sync In Prog	Synchronised	Stopped	Failback
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the R1 Symmetrix array is running an Enginuity level lower than 5874.
2. Not allowed if the R2 owes data to the R1.
3. Not allowed if the Rcopy session has front end zero detect.
4. If R2 owes data to the R1, not allowed if the R1 Symmetrix is running an Enginuity level less than 5874.

Rcopy Session on the R2 side

This section lists the allowable SRDF operations with their applicable Rcopy states when there is an Rcopy session on the R2.

R2 is part of an Rcopy PUSH

Table 65 identifies the allowable SRDF actions with their applicable Rcopy states when there is an Rcopy PUSH session on the R2.

Table 65 Allowable SRDF operations with Rcopy PUSH session on the R2

Rcopy state: / SRDF control operation:	None	Create in Progress	Created	Copy In Progress	Copy on Write	Copied	Recreate In Progress	Recreated	Terminate In Progress	Failed	Invalid	Verify In Progress	Restored	Restored In Prog	Precopy	Sync In Prog	Synchronised	Stopped
createpair -establish	Y																	
createpair -restore	Y																	
createpair -invalidate R1	Y																	
createpair -invalidate R2	Y																	
createpair -format	Y																	
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 65 Allowable SRDF operations with Rcopy PUSH session on the R2 (continued)

Rcopy state: / SRDF control operation:	None	Create in Progress	Created	Copy In Progress	Copy on Write	Copied	Recreate In Progress	Recreated	Terminate In Progress	Failed	Invalid	Verify In Progress	Restored	Restored In Prog	Precopy	Sync In Prog	Synchronised	Stopped
swap	Y																	
half_swap	Y																	
swap -refresh R1	Y																	
swap -refresh R2	Y																	
establish	Y		Y[1]			Y[1]		Y[1]					Y[1,2,3]					
establish -full	Y		Y[1]			Y[1]		Y[1]					Y[1,2,3]					
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,3]	Y[3]	Y[1]	Y[1]	Y[1]	Y[1]
restore -full	Y	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1]	Y[1,3]	Y[3]	Y[1]	Y[1]	Y[1]	Y[1]
update	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y[3]	N	Y	Y	Y	Y
failback	Y																	
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y																	
failover -restore	Y																	
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R2	Y																	
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y		Y[1]			Y[1]		Y[1]					Y[1,2,3]					
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 65 Allowable SRDF operations with Rcopy PUSH session on the R2 (continued)

Rcopy state: / SRDF control operation:	None	Create in Progress	Created	Copy In Progress	Copy on Write	Copied	Recreate In Progress	Recreated	Terminate In Progress	Failed	Invalid	Verify In Progress	Restored	Restored In Prog	Precopy	Sync In Prog	Synchronised	Stopped
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1. Not allowed if the R2 Symmetrix array is running an Enginuity level lower than 5874.
2. Not allowed if donor update specified.
3. Not allowed if the Rcopy session has front end zero detect.

R2 is part of an Rcopy PULL

Table 66 identifies the allowable SRDF actions with their applicable Rcopy states when there is an Rcopy PULL session on the R2.

Table 66 Allowable SRDF operations with Rcopy PULL session on the R2

Rcopy State: / SRDF control operation:	None	Create in Progress	Created	Copy in Progress	Copy on Access	Copied	Terminate In Progress	Failed	Invalid	Verify In Progress	Sync In Prog	Synchronised	Stopped	Failback
createpair -establish	Y													
createpair -restore	Y													
createpair -invalidate R1	Y													
createpair -invalidate R2	Y													
createpair -format	Y													
deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_deletepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
half_movepair	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
swap	Y													
half_swap	Y													
swap -refresh R1	Y													
swap -refresh R2	Y													
establish	Y													
establish -full	Y													
split	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
restore	Y													
restore -full	Y													
update	Y													
failback	Y													
failover	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
failover -establish	Y													
failover -restore	Y													
invalidate -R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
invalidate -R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 66 Allowable SRDF operations with Rcopy PULL session on the R2

Rcopy State: / SRDF control operation:	None	Create in Progress	Created	Copy in Progress	Copy on Access	Copied	Terminate In Progress	Failed	Invalid	Verify In Progress	Sync In Prog	Synchronised	Stopped	Failback
merge	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
msc_cleanup	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
not_ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
ready R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
refresh R2	Y													
suspend	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
resume	Y													
rw_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
rw_enable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
write_disable R2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_dse	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_devpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_pace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
activate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
deactivate -rdfa_wpace_exempt	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y