



# EMC<sup>®</sup> Solutions Enabler Symmetrix Array Controls CLI

Version 7.5

## Product Guide

P/N 300-014-875  
REV 01

Copyright ©2012 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.

EMC2, EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

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 Managing Configuration Changes

Introduction .....	22
Configuration change session overview .....	24
Making configuration changes safely .....	28
Configuration operations supported .....	30
Setting Symmetrix metrics .....	33
Planning system protection configurations .....	36
Managing external disk groups .....	42
Deleting a disk group .....	42
Managing devices .....	43
Creating devices .....	46
Copying a device .....	52
Managing metadevices .....	53
Automatic metadevice creation .....	56
Managing spare disks .....	62
Configuring thin metadevices .....	63
Converting devices .....	64
Setting device attributes .....	70
Setting device geometry .....	72
Setting device identifiers .....	75
Deleting devices .....	77
Managing device reservations .....	78
Managing device pools .....	83
Creating a device pool .....	84
Renaming a device pool .....	85
Monitoring a device pool .....	86
Viewing a device pool .....	87
Mapping devices .....	88
Mapping CKD devices .....	90
Mapping a range of devices .....	90
Copying the mapping of a range of devices .....	90
Assigning PAV alias addresses to mapped devices .....	91
Removing PAV alias addresses from mapped devices .....	91
Unmapping devices .....	92
Unmapping a range of devices .....	92
Configuring dynamic RDF-capable devices .....	93
Enabling dynamic SRDF on the device .....	94
Setting SRDF group attributes .....	95
Adding/removing SRDF mirrors .....	97
Swapping RA groups .....	100
Converting directors .....	101
Setting port characteristics .....	103
Viewing port flags .....	107

<b>Chapter 2</b>	<b>Configuring Virtual Provisioning</b>	
	Virtual Provisioning overview .....	110
	Virtual Provisioning (VP) compression .....	111
	Creating DATA devices .....	112
	Creating thin devices (TDEVs) .....	113
	Restrictions when preallocating the size.....	114
	Example of create/bind .....	114
	Creating thin pools.....	116
	Binding thin devices .....	118
	Restrictions when preallocating the size.....	118
	Unbinding thin devices .....	119
	Rebinding thin devices.....	120
	Restrictions for rebinding thin devices.....	120
	Setting pool attributes .....	121
	Allocating space on a thin device .....	123
	Space reclamation .....	124
	Stop reclamation.....	124
	Stop free .....	124
	Stopping a background operations on thin devices .....	125
	Stop free .....	125
	Stop allocate.....	125
	Draining a thin pool .....	126
	Deactivating devices in a thin pool .....	128
	Starting the drain on DATA devices.....	128
	Activating devices in a thin pool.....	128
	Stopping the drain on DATA devices .....	129
	Automated pool rebalancing .....	129
	Stopping pool rebalancing .....	130
	Verifying the pool and device states .....	131
	Viewing thin pools .....	131
 <b>Chapter 3</b>	 <b>Masking Devices with Auto-provisioning Groups</b>	
	Auto-provisioning Groups.....	136
	How it works .....	136
	Discovering host HBAs .....	138
	Creating groups and views .....	138
	Creating storage groups .....	138
	Creating port groups.....	140
	Creating initiator groups.....	141
	Creating a masking view.....	142
	Managing masking views .....	144
	Verifying the database .....	144
	Deleting masking views.....	145
	Naming groups and views .....	145
	Backing up and restoring views .....	146
	Unused storage groups .....	146
	Copying groups and views.....	148
	Managing storage groups.....	149
	Adding devices .....	149
	Setting Host I/O Limits .....	151
	Removing devices .....	152
	Deleting storage groups .....	153
	Managing port groups.....	154
	Adding ports .....	154

Removing ports .....	154
Deleting port groups.....	154
Locking down a Fibre Channel ID .....	155
Managing initiator groups .....	156
Adding initiators .....	156
Setting HBA flags .....	159
Replacing an HBA.....	160
Renaming an HBA .....	161
Using CHAP authentication.....	161
Displaying Auto-provisioning Group information.....	162
Displaying masking views .....	162
Viewing group details.....	163
Viewing device assignments .....	165
Listing initiator group devices .....	166
Viewing the HBA alias name.....	167
Auto-provisioning Groups examples .....	168

## Chapter 4

### Fully Automated Storage Tiering

What is FAST?.....	170
FAST overview .....	172
FAST VP overview .....	175
FAST VP example.....	177
Managing Symmetrix tiers .....	178
Creating disk group tiers .....	180
Creating virtual pool tiers .....	181
Listing and showing tier information .....	184
Managing policies.....	188
Creating a FAST policy .....	189
Adding a tier to a policy.....	190
Removing a tier from a policy .....	191
Deleting a policy .....	191
Listing and showing policies .....	192
Managing storage groups.....	193
Creating storage groups .....	195
Associating a storage group to a policy .....	196
Reassociating a storage group to a new policy.....	197
Disassociating a storage group from a policy.....	197
Modifying a storage group.....	198
Displaying storage group associations .....	198
Managing the FAST controller .....	200
Enabling and disabling FAST .....	201
Displaying the FAST state .....	202
Setting FAST control parameters.....	203
Displaying FAST plans .....	207
Approving a FAST plan.....	209
Declining a FAST plan .....	210
Displaying FAST information.....	211
FAST activity history .....	211
FAST reports.....	214
Compliance report.....	214
Technology demand reports .....	216

<b>Chapter 5</b>	<b>Federated Tiered Storage</b>	
	What is FTS?.....	220
	Using eDisks .....	221
	Encapsulation .....	221
	External provisioning .....	221
	Geometry of encapsulated devices.....	222
	Geometry limited devices.....	222
	Adding an eDisk.....	224
	Removing an eDisk.....	225
	Displaying external disk information .....	225
	Solutions Enabler FTS support.....	226
 <b>Chapter 6</b>	 <b>Enhanced Virtual LUN Technology</b>	
	Overview.....	230
	VP compression guidelines .....	232
	Creating virtual LUN migration sessions .....	235
	Migrating to configured space .....	236
	Migrating to unconfigured space .....	239
	Migrating to a thin pool.....	240
	Managing virtual LUN migration sessions .....	242
	Listing sessions .....	243
	Querying sessions.....	244
	Verifying sessions .....	246
	Terminating sessions .....	248
	Viewing the audit log.....	248
	Virtual LUN migration examples .....	249
 <b>Chapter 7</b>	 <b>Managing Time Windows</b>	
	Managing time windows .....	252
	Converting time windows .....	254
	Adding time windows.....	256
	Removing time windows.....	257
	Displaying time windows.....	258
	Time window examples .....	261
 <b>Chapter 8</b>	 <b>Managing Quality of Service</b>	
	Overview.....	266
	Setting copy pace and priority .....	266
	Setting host I/O priority .....	270
	Setting and resetting LRU parameters .....	271
	Setting RDF director workload distribution.....	272
	Managing dynamic cache partitions .....	275
	QoS examples.....	280
 <b>Chapter 9</b>	 <b>Optimizing Array Performance</b>	
	What is Symmetrix Optimizer? .....	284
	How does Symmetrix Optimizer work? .....	287
	Symmetrix Optimizer operations using SYMCLI.....	293
	Setting control parameters.....	295
	Setting advanced parameters.....	295
	Setting time windows.....	296

	Setting the swap priority for devices.....	296
	Setting manual swap lists .....	297
	Reviewing Optimizer plans .....	297
	Approving a swap list .....	306
	Performing rollbacks .....	307
	Reading Optimizer logs .....	307
	Migrating devices.....	308
<b>Chapter 10</b>	<b>Performing Double Checksum Operations</b>	
	Overview.....	312
	Implementing EMC Double Checksum for Oracle .....	313
	Implementing Generic SafeWrite for generic applications .....	317
	Syntax and examples .....	322
<b>Chapter 11</b>	<b>Device Masking</b>	
	Device masking overview .....	324
	Supported topologies .....	327
	Using the device masking database .....	329
	Configuring device masking .....	329
	Discovering host HBAs .....	332
	Adding masked devices .....	334
	Removing masked devices .....	336
	Maintaining the device masking database.....	338
	Viewing the VCMDB.....	340
	Managing HBA initiators.....	346
	Managing the Fibre Channel-to-host interface .....	348
<b>Chapter 12</b>	<b>Device Masking: iSCSI Setup</b>	
	iSCSI authentication overview .....	358
	iSCSI CHAP authentication: Enginuity version 56xx .....	358
	iSCSI CHAP authentication: Enginuity version 57xx .....	359
	iSCSI software driver configuration.....	363
	RADIUS server configuration .....	376
<b>Chapter 13</b>	<b>Managing Network IPsec</b>	
	Introduction.....	380
	IPsec and Symmetrix.....	381
	SYMCLI IPsec management component .....	382
	Viewing policy information.....	383
	Command file scripting for creating/modifying policies .....	387
	Committing a policy-scripted command file .....	392
	Deleting a policy-scripted command file .....	392

<b>Part 2</b>	<b>Operational Examples</b>	
<b>Appendix A</b>	<b>Configuration Change Examples</b>	
	Creating devices.....	396
	Mapping devices.....	398
	Setting front-end port attributes.....	402
	Configuring concurrent SRDF devices .....	406
	Updating a disk label with emulated device geometry .....	412
	Configuring Virtual Provisioning .....	420
	Creating a pool.....	420
	Adding DATA devices .....	421
	Binding thin devices to the pool .....	423
	Listing thin devices in a pool .....	424
	Showing all the pool details .....	425
	Automatically creating metadevices .....	427
	Setting port flags on a front-end director .....	430
	Removing port flags from a front-end director.....	431
<b>Appendix B</b>	<b>Auto-provisioning Groups Example</b>	
	Creating and displaying children and parent storage groups.....	434
	Showing parent storage groups.....	434
	Showing children storage groups .....	435
	Creating and displaying port groups .....	435
	Displaying port group information .....	435
	Creating and showing initiator groups .....	436
	Displaying initiator group information .....	436
	Creating and displaying masking views .....	437
	Showing masking view information .....	437
<b>Appendix C</b>	<b>FAST Output Examples</b>	
	Showing associations for mixed storage groups using -all .....	440
	Showing associations for standard tiers .....	441
	Listing the FAST plan for disk groups with the -v option .....	442
	Listing the history with the -v option.....	443
	Listing the technology demand for DP tiers.....	444
	Listing the technology demand for VP tiers using -v .....	446
	Listing the technology demand for DP tiers using -v.....	449
	Listing the technology demand using V71 mode .....	451
<b>Appendix D</b>	<b>Virtual LUN Migration Example</b>	
	Virtual LUN migration example .....	456
<b>Appendix E</b>	<b>Updating the Host</b>	
	Sun Solaris systems.....	466
	HP-UX systems.....	466
	IBM AIX systems .....	466
	HP Tru64 UNIX systems .....	467
	Windows systems .....	467
	HP Open VMS systems .....	468



<b>Appendix F</b>	<b>SIU: Overview and Management</b>	
	Introduction .....	470
	Configuring a VMware virtual server .....	472
	Viewing detailed information about disks .....	474
	Mounting and unmounting disks.....	475
	Manipulating disk signatures .....	477
	Updating partition tables .....	477
	Flushing cache buffers to disk.....	478
	Device masking operations .....	478
	Using volume flags.....	479
	Performing typical storage operations .....	480
	SIU examples.....	485
	XML examples.....	486
<b>Appendix G</b>	<b>Managing Legacy Time Windows</b>	
	Setting time windows.....	488



# FIGURES

	Title	Page
1	Symmetrix parity RAID mirror scheme .....	39
2	Simple SRDF device configuration .....	40
3	Various SRDF-connected device configurations with BCVs .....	41
4	Concatenated metadevices .....	54
5	Striped data .....	55
6	Data allocation for thin devices .....	110
7	Draining a thin pool.....	127
8	Masking view overview.....	137
9	Storage group SG_1 .....	140
10	Port group PG_1 .....	141
11	Initiator group IG_1 .....	142
12	Masking view MV_1 .....	143
13	Two masking views .....	144
14	FAST component associations .....	171
15	Sample Symmetrix array - before FAST DP configuration .....	172
16	FAST DP configuration example .....	173
17	FAST VP configuration example .....	177
18	FAST policy overview .....	188
19	FTS configuration .....	220
20	Initial state of RAID 1 device with SRDF.....	232
21	Synchronizing the RAID 5 group .....	233
22	RAID 5 group with SRDF.....	233
23	Time window weekly format .....	253
24	Symmetrix Optimizer application architecture .....	284
25	Swap steps using one DRV .....	291
26	Identify volumes to swap.....	291
27	Copy volume to DRV .....	292
28	Copy DRVs to new locations .....	292
29	DRVs are split.....	292
30	Generic SafeWrite list output.....	321
31	Device masking solution with WWN initiators .....	325
32	Point-to-point and multi-initiator topologies.....	328
33	Native iSCSI topology .....	328
34	Device masking components.....	331
35	IPsec network tunnel with Symmetrix-to-Symmetrix endpoints .....	382
36	IPsec network tunnel with iSCSI host-to-Symmetrix endpoints.....	382
37	Device view prior to migration session .....	456
38	Device view after the migration session.....	462



# TABLES

	Title	Page
1	Configuration change controls.....	23
2	Configuration change operations .....	30
3	Virtual mirror positions.....	36
4	Mirror set protection schemes .....	37
5	Steps to create a device .....	43
6	Maximum device sizes by Enginuity version .....	46
7	Metadevice sizes by Enginuity version for auto_meta .....	54
8	Steps to create an SRDF BCV metadevice .....	57
9	Metadevice data preservation .....	61
10	Thin device conversions .....	67
11	SCSI protocol port flags.....	104
12	Fibre protocol port flags .....	106
13	Supported thin metadevice features.....	115
14	symaccess restore behavior .....	146
15	FAST version differences .....	170
16	FTS support across Solutions Enabler features .....	226
17	Virtual LUN requirements by features .....	231
18	Source and target compression guidelines .....	232
19	Action descriptions for symmigrate command .....	235
20	Audit log session information.....	248
21	symqos command descriptions.....	266
22	Optimizer strategies for determining swaps.....	289
23	Symmetrix Optimizer CLI command .....	293
24	Optimizer control parameters .....	295
25	New fields in Optimizer display .....	302
26	EMC Double Checksum for Oracle functionality by Enginuity level .....	316
27	symchksum syntax and descriptions .....	322
28	LUN base/offset scenarios for multiple hosts with skip holes .....	351
29	Host platforms and interface configuration flags .....	354
30	SIU command summary .....	471
31	Optimizer time window parameters .....	488



# 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 the EMC online support website. Check the EMC online support website to ensure that you are using the latest version of this document.

---

## Purpose

This document describes how to configure and use EMC Solutions Enabler.

## Audience

This manual provides both guide and reference information for command-line users and script programmers that focus on configuring and managing storage on Symmetrix arrays.

## Related documentation

The following EMC publications provide additional information:

- ◆ *EMC Solutions Enabler Security Configuration Guide*
- ◆ *EMC Solutions Enabler Installation Guide*
- ◆ *EMC Solutions Enabler Symmetrix CLI Command Reference*
- ◆ *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix Migration CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix SRM CLI Product Guide*
- ◆ *EMC Solutions Enabler Symmetrix CLI Quick Reference*
- ◆ EMC host connectivity guides for [your operating system]

## Conventions used in this document

EMC uses the following conventions for special notices:

---

**Note:** A note presents information that is important, but not hazard-related.

---

### **IMPORTANT**

---

An important notice contains information essential to software or hardware operation.

---

## Typographical conventions

EMC uses the following type style conventions in this document:

Normal	Used in running (nonprocedural) text for: <ul style="list-style-type: none"> <li>Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus</li> <li>Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities</li> <li>URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications</li> </ul>
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: <ul style="list-style-type: none"> <li>Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus</li> <li>What the user specifically selects, clicks, presses, or types</li> </ul>
<i>Italic</i>	Used in all text (including procedures) for: <ul style="list-style-type: none"> <li>Full titles of publications referenced in text</li> <li>Emphasis, for example, a new term</li> <li>Variables</li> </ul>
Courier	Used for: <ul style="list-style-type: none"> <li>System output, such as an error message or script</li> <li>URLs, complete paths, filenames, prompts, and syntax when shown outside of running text</li> </ul>
<b>Courier bold</b>	Used for specific user input, such as commands
<i>Courier italic</i>	Used in procedures for: <ul style="list-style-type: none"> <li>Variables on the command line</li> <li>User input variables</li> </ul>
< >	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 the EMC Online Support site as described next.

---

**Note:** To open a service request through the EMC Online Support site, 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 <https://support.EMC.com>, as directed on your License Authorization Code (LAC) letter emailed 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 the Online Support site, contact EMC's worldwide Licensing team at [licensing@emc.com](mailto: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.



# PART 1

## Concepts and Procedures

The Concepts and Procedures part of this product guide provides conceptual information and describes how to perform various types of control operations on Symmetrix devices using the SYMCLI commands of the EMC Solutions Enabler software. These concepts and procedures are described in the subsequent chapters, as follows:

[Chapter 1, “Managing Configuration Changes,”](#) describes configuration concepts and how to manage various types of configurations using the configuration commands of SYMCLI.

[Chapter 2, “Configuring Virtual Provisioning,”](#) explains how to set up and use Virtual Provisioning.

[Chapter 3, “Masking Devices with Auto-provisioning Groups,”](#) describes storage provisioning concepts and how to confine host access to Symmetrix devices using the `symaccess` command of SYMCLI.

[Chapter 4, “Fully Automated Storage Tiering,”](#) describes Fully Automated Storage Tiering (FAST) concepts and how use the `symfast` and `symtier` commands of SYMCLI.

[Chapter 5, “Federated Tiered Storage,”](#) describes Federated Tiered Storage (FTS) concepts and explains how to configure and manage FTS using the SYMCLI.

[Chapter 6, “Enhanced Virtual LUN Technology,”](#) explains how to perform virtual LUN migrations and use the `symmigrate` command of SYMCLI.

[Chapter 7, “Managing Time Windows,”](#) explains how to add, remove, convert, and display time windows using the `symtw` command.

[Chapter 8, “Managing Quality of Service,”](#) describes QoS concepts and how to manage the Quality of Service (QoS) on devices in your storage environment using the `symqos` command of SYMCLI.

[Chapter 9, “Optimizing Array Performance,”](#) describes Symmetrix Optimizer concepts and how to use the `symoptmz` command of SYMCLI.

[Chapter 10, “Performing Double Checksum Operations,”](#) describes Double Checksum concepts and how to use the `symchecksum` command of SYMCLI.

[Chapter 11, “Device Masking,”](#) describes device masking concepts and how to confine host access to Symmetrix devices using the device masking commands of SYMCLI.

[Chapter 12, “Device Masking: iSCSI Setup,”](#) describes how to configure your iSCSI driver software and authentication information.

[Chapter 13, “Managing Network IPsec,”](#) describes the IPsec standard and how to manage IPsec network policies, using the IPsec component of SYMCLI.



# CHAPTER 1

## Managing Configuration Changes

This chapter describes configuration change concepts and how to perform change operations using the Configuration Change component of the SYMCLI. The chapter covers the following topics:

◆ Introduction .....	22
◆ Configuration change session overview .....	24
◆ Making configuration changes safely.....	28
◆ Configuration operations supported.....	30
◆ Setting Symmetrix metrics.....	33
◆ Planning system protection configurations .....	36
◆ Managing external disk groups .....	42
◆ Managing devices .....	43
◆ Managing metadevices .....	53
◆ Managing spare disks .....	62
◆ Configuring thin metadevices .....	63
◆ Converting devices.....	64
◆ Setting device attributes .....	70
◆ Setting device geometry.....	72
◆ Setting device identifiers .....	75
◆ Deleting devices .....	77
◆ Managing device reservations.....	78
◆ Managing device pools .....	83
◆ Mapping devices.....	88
◆ Mapping CKD devices.....	90
◆ Unmapping devices .....	92
◆ Configuring dynamic RDF-capable devices.....	93
◆ Setting SRDF group attributes .....	95
◆ Adding/removing SRDF mirrors .....	97
◆ Swapping RA groups .....	100
◆ Converting directors.....	101
◆ Setting port characteristics .....	103

# Introduction

The SYMCLI configuration change command, `symconfigure`, is used to perform control operations on Symmetrix® arrays, and the array devices, tiers, groups, directors, and ports. Some of the Symmetrix array controls include setting array-wide metrics, and determining what type of devices the array will support, such as RAID 6 devices. Device controls include creating devices, mapping and masking devices, and configuring device pools. The `symconfigure` command is also used for reserving devices and releasing device reservations.

You can invoke `symconfigure` from the local host to make configuration changes to a locally-connected Symmetrix array, or to an RDF-linked Symmetrix array.

There are three ways to process `symconfigure` commands, as follows:

- ◆ The most common way to use `symconfigure` is placing commands, terminated with a semicolon (;), in a command file. An example follows:

```
symconfigure commit -sid 3160 -file unmap_dev.cmd
```

Where `unmap_dev.cmd` contains:

```
unmap dev 020:024 from dir ALL:ALL;
```

---

**Note:** When using the `symconfigure -file` option, text files can have a maximum comment of 512 characters on Windows. Make sure the comment line does not exceed 512 characters.

---

- ◆ Optionally, for UNIX platforms, you can redirect a number of screen entries to `stdin` to save keystroke entries and not deal with the command file.

For example, to prepare a chain of `symconfigure` commands on the screen to be redirected to `stdin`, use the following form:

```
symconfigure -sid SymmID prepare <<DELIM
create dev count=3 size=3200 cyl,
emulation=FBA, config=2-Way-Mir,
create dev count=1, size = 3200 cyl,
emulation=FBA, config=unprotected;
DELIM
```

- ◆ A similar screen method can be used with the `-cmd` option. With this option, the commands that would normally be put in a command file are enclosed in quotes. A command can run over many lines, but you cannot press **Enter**. An example follows:

```
symconfigure -sid 256 -cmd "create dev count=3, size = 3200 cyl,
emulation=FBA, config=2-Way-Mir;create dev count=1, size = 3200 cyl,
emulation=FBA, config=unprotected;" -v -nop preview
```

This section describes the `symconfigure` arguments, and the remaining chapter explains the command operations that can be written in a command file.

Configuration changes that are submitted to the array are processed in a session. The `symconfigure` command file format contains various command entries terminated with a semicolon (;). Multiple changes can be made in one session, but all changes must fall into

one complete operation. For example, creating a device, adding the device to a device pool, and enabling the device state. Additional command file examples are provided with the actions described later in this chapter.

Table 1 defines the `symconfigure` command arguments:

**Table 1** Configuration change controls

symconfigure option	Definition
abort	Gains control of an existing session to abort it and free any held locks.
commit	Attempts to apply the changes defined in the command file into the specified Symmetrix array.
list	Lists the relevant details, depending on the option: <ul style="list-style-type: none"> <li>• <code>-freespace</code> shows the free physical disk space within the Symmetrix array as it can be used to create new Symmetrix devices for different emulation modes. Free disk space on unformatted disks is shown as available for all emulation modes. If a physical disk has been partially used to create a device, that device is considered formatted and the rest of the available space can only be used for devices of the same emulation mode.</li> <li>• <code>-v</code> displays configuration information that is not stored in the SYMAPI database and that needs to be retrieved directly from the configuration server.</li> <li>• <code>-reserve</code> shows a summary of all reservations.</li> </ul>
prepare	Validates the syntax and correctness of the operations. Verifies the validity of the command file changes and their appropriateness for the specified Symmetrix array. The prepare action has no function for pool sessions.
preview	Ensures the command file syntax is correct. Verifies the validity of the command file changes.
query	Returns information about the status of a configuration change session.
release	Releases the specified device reservation.
reserve	Processes the command file to reserve the indicated devices and displays the resulting reserve ID.
show	Shows the details of the specified device reservation.
verify	Verifies that the configuration currently running in the specified Symmetrix array complies with the requirements for host-based configuration changes.

Refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference* for the complete man page description of the `symconfigure` command.

## Mixed array environments

Before issuing a `symconfigure` command to a Symmetrix array containing both open system and mainframe devices, you must verify that the mainframe Missing Interrupt Handler (MIH) period for *each* mainframe server attached to the Symmetrix array is set to at least 2 minutes.

To view the current MIH period, use the z/OS command `D IOS,MIH`, and note the value for the DASD device class (for example, `DASD=1:30`).

To change the MIH period, use the z/OS command `SETIOS MIH,DASD=mm:ss`, where `mm:ss` is a period of time in minutes and seconds. Then use the z/OS command `D IOS,MIH` to verify your changes.

Once you have completed your configuration change session, use the `SETIOS MIH, DASD=mm:ss` command to set the MIH period back to its original value.

For more information, refer to your mainframe system administrator.

## Configuration change session overview

Solutions Enabler 7.0 and higher, running on the Symmetrix VMAX™ Family with Enginuity™ handles configuration changes differently from previous versions of Solutions Enabler. The following is a list of changes in the configuration change feature of Solutions Enabler 7.0 (and higher) with the Enginuity operating environment for Symmetrix arrays:

- ◆ *Concurrent Provisioning:* Up to four concurrent configuration change sessions are allowed to run at the same time, when they are non-conflicting. This means that multiple parallel configuration change sessions can run at the same time as long as the changes do not include any conflicts on the following:
  - Device back-end port
  - Device front-end port
  - Device
- ◆ Devices and metadevices can be created and mapped in one command. This eliminates the need to perform multiple configuration changes when provisioning new storage.
- ◆ The Symmetrix array manages its own less restrictive device locking.
- ◆ A session ID identifies each running session on the array.

---

**Note:** New features in Enginuity 5874 are not supported on arrays running Enginuity 5773 and earlier unless noted.

---

A session acting on a specified `symconfigure` command file can be verified and checked using the `preview` and `prepare` arguments. The `commit` argument performs these same checks, but then attempts to execute the specified action.

Not all stages are always executed. Use discretion when controlling which stages are to be completed to allow checking and debugging of the command file before the changes are implemented.

### Preview

The `preview` argument verifies the syntax and correctness of each individual change defined, their correctness as a set, and then terminates the session without change execution (correct within the realm of the host and valid as a possible Symmetrix configuration).

### Prepare

The `prepare` argument performs the preview checks, validates the change operation (devices are in correct state, etc.), and also verifies the appropriateness of the resulting configuration definition against the current state of the Symmetrix arrays. The argument



then terminates the session without attempting to make the configuration change. For Symmetrix VMAX Family arrays running Enginuity 5874, the `prepare` argument performs the same actions as the `preview` argument.

---

**Note:** The `prepare` argument has no function for SAVE device pool sessions.

---

An example of system output for `symconfigure prepare` command follows:

```
symconfigure -sid 256 -file script.mkdevice -v -nop prepare
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000187940256
{
  create dev count=1, size=3200, emulation=FBA,
  config=2-Way Mir, mvs_ssid=0000;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Submitting configuration changes.....Submitted
Validating configuration changes.....Validated.

New symdev: 038B
Initiating PREPARE of configuration changes.....Queued.
PREPARE requesting required resources.....Obtained.
Step 013 of 017 steps.....Executing.
Step 014 of 017 steps.....Executing.
Local: PREPARE.....Done.
Closing configuration change request.....Closed.
Terminating the configuration change session.....Done.
```

The configuration change session has completed successfully.

## Commit

The `commit` argument completes all checks and verifications, and then attempts to make the requested configuration changes in the specified Symmetrix array.

## Query

After you commit a command file for execution, it takes a few minutes to complete execution. You can check the status of any configuration change session using the `query` argument.

To check the status of the change session for the next 2 minutes, every 10 seconds on Symmetrix 12345, enter:

```
symconfigure -sid 12345 query -i 10 -c 12
```

The `query` command is useful in SRDF environments, where a change to a local Symmetrix array on one host results in a corresponding change to a remote Symmetrix array. The System Manager of a host connected to the remote Symmetrix array can monitor the progress of the change. A `query` is also helpful at sites where the Symmetrix Optimizer is also modifying a configuration by rearranging the placement of data.

The `query` command can also be used to show if there is a configuration session running.

The following is an example of the `symconfigure query` command on a Symmetrix array running Enginuity 5874:

**symconfigure -sid 207 query**

A Configuration Change query is in progress. Please wait...

A Configuration Change operation is in progress. Please wait...

Establishing a monitoring session.....Established.

Session ID : 158209 (0x00026a01)

```
{
  The session changes are in the class of: Modifying symmetrix constraints
  {
    set symmetrix auto_meta_config=Concatenated;
  }
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : NA
The Process ID that initiated the configuration change : 31831
The session length                                   : 12 secs
The session status                                    : Running
```

```
The last action requested was: COMMIT
The state of the last action is: Running
Step 46 of 65 steps.....Executing.
```

}

Session ID : NA

```
{
  The session changes are in the class of: NA
  The Application that initiated the configuration change : EMC Internal
  The Host that initiated the configuration change       : NA
  The Process ID that initiated the configuration change : NA
  The session length                                   : 56 secs
  The session status                                    : Running
}
```

Session ID : 201 (0x000000c9)

```
{
  The session changes are in the class of: mapping devices
  {
    map dev 100 to dir 15D:0, lun = F0;
  }
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : NA
The Process ID that initiated the configuration change : 244
The session length                                   : 30 secs
The session status                                    : Running
```

```
The last action requested was: COMMIT
The state of the last action is: Running
Step 57 of 105 steps.....Executing.
```

}

Terminating the monitoring session.....Done.

## Abort

You can abort a configuration session. To abort a change session on Symmetrix 12345, enter:

**symconfigure -sid 12345 abort**

Because changes made in the SRDF operations class will initiate actions on the local Symmetrix and remote Symmetrix arrays, it might become necessary to abort processing on a remote Symmetrix array.

At some point during `commit` processing, a *point of no return* is reached. Any attempt to abort will be denied once processing has reached this point.

## Session ID

Add the `-session_id` option to the `symconfigure abort` command, to differentiate between multiple running sessions. To abort the change session 100 on a Symmetrix array 343, enter:

```
symconfigure -sid 343 abort -session_id 100
```

If the `session_id` is not known, running the `symconfigure -sid SymmID abort` command will display each running session and prompt for the session ID, as shown in the following example:

```
symconfigure -sid 343 abort
```

```
A Configuration Change operation is in progress. Please wait...
```

```
Looking for an existing configuration sessions.....Established.
```

```
Session ID : 100 (0x00000064)
```

```
{
  {
    set symmetrix auto_meta = Disabled;
  }
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : LQ283
The Process ID that initiated the configuration change  : 15023
The session length                                   : 28 secs
The session status                                    : Running
```

```
}
```

```
Session ID : 105 (0x00000069)
```

```
{
  {
    map dev 100 to dir 15D:0, lun = F0;
  }
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : LQ283
The Process ID that initiated the configuration change  : 244
The session length                                   : 30 secs
The session status                                    : Running
```

```
}
```

```
Session ID : 2100
```

```
{
  {
    map dev 104 to dir 2C:0, lun = 0F;
  }
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : LQ283
The Process ID that initiated the configuration change  : 248
The session length                                   : 22 secs
The session status                                    : Running
```

```

}

Please enter the ID of the session you wish return to cancel abort: 100

Session ID : 100 (0x00000064)
{
  {
    set symmetrix auto_meta = Disabled;
  }

  The Application that initiated the configuration change : SYMCONFIGURE
  The Host that initiated the configuration change       : LQ283
  The Process ID that initiated the configuration change : 15023
  The session length                                   : 28 secs
  The session status                                    : Running
}

Abort the configuration change session 1 for Symmetrix unit 000190300343 (y/[n])? y
Aborting configuration changes.....Aborted.
Terminating the configuration change session.....Done.

```

## Making configuration changes safely

Before making configuration changes, you must thoroughly understand your Symmetrix configuration. The following guidelines help to establish safe disciplines as the changes you make can easily impact stored data:

- ◆ Verify that the current Symmetrix configuration is a viable configuration for host-initiated configuration changes.

To verify your current Symmetrix configuration is ready for changes, enter:

```
symconfigure verify -sid SymmID
```

- ◆ Before creating new Symmetrix devices, check for free physical disk space using the following command:

```
symconfigure list -freespace [-units CYL | MB] -sid SymmID
```

Free disk space on unformatted disks is shown as available for all emulation modes. New Symmetrix devices will be created first on physical disks that have no prior allocations, causing these disks to be committed to that emulation type.

Examine the distribution of free space across formatted disks to see if the desired mirroring can be provided using one of the following commands:

```
symdev list -sid SymmID -da all -space
or
symdisk list -sid SymmID
```

- ◆ No configuration change is activated in the Symmetrix system until you `commit` the action.

Some classes of change operations may or may not impact current I/O. When possible, before you `commit` any action, stop I/O activity on the affected Symmetrix devices.

If required, set the impacted devices for change to be Not Ready using the following command:

```
symdg -g DgName not_ready [LdevName [LdevName... ]]
```

---

**Note:** If I/O activity on an affected Symmetrix device occurs before or during a `commit` action, the `commit` action might fail. At the very least, heavy I/O activity on unaffected devices will impact how long it takes to commit changes.

---

- ◆ Ensure that all your critical data is preserved and safe when creating new or changing device configurations. Do not store data on any device that is not mirrored, or RAID-protected.

All final configurations and device attribute adjustments must meet certain open systems guidelines detailed in the E-Lab™ Interoperability Navigator, which can be viewed at <http://elabnavigator.EMC.com>.

Contact the EMC® Customer Support Center for assistance in reverting to your previous configuration should there be unforeseen problems with the new configuration.

- ◆ After committing a `symconfigure` mapping operation, you must update the device mapping information within the host system environment. Attempting host activity with a device after it has been removed or altered, but before you have updated the host's device information, can cause host errors.

To update your hosts, run the utilities specified for your platform as described in [Appendix E, "Updating the Host."](#) The `symcfg discover` command should then be issued to update the SYMAPI database with the new device mapping information. After the host environment is updated, I/O activity can resume with the Symmetrix device.

## Change Tracker session priority

The following `symconfigure` operations will be blocked when there is an active Change Tracker session on the related devices:

- ◆ Converting to VCMDB device
- ◆ Forming and dissolving a metadvice
- ◆ Adding metamember to a metadvice
- ◆ Removing metamembers from a metadvice
- ◆ Converting a metadvice from concatenated to striped
- ◆ Deleting a device
- ◆ Unbinding a thin device

## Enable SYMAPI environment option

Before executing any of the class changes, the environment option `SYMAPI_CTRL_OF_NONVISIBLE_DEVS` in the `options` file might need to be enabled or removed. This option is normally enabled, which is required for many of the configuration changes available.

**Note:** A nonvisible device is a device that is not mapped to your host and/or does not have a physical host device name.

The editable `options` file is located in the SYMAPI configuration directory:

`/var/symapi/config` (for UNIX)  
`C:\Program Files\EMC\Symapi\config` (for Windows)

To enable `SYMAPI_CTRL_OF_NONVISIBLE_DEVS`, set:

```
SYMAPI_CTRL_OF_NONVISIBLE_DEVS=ENABLE
```

The options file chapter in the *EMC Solutions Enabler Symmetrix CLI Command Reference* explains how to edit the options file.

## Configuration operations supported

The Enginuity requirements for the configuration change operations are listed in [Table 2](#).

**Table 2** Configuration change operations (page 1 of 4)

Configuration change operation	5671	5771	5772	5773	5874	5875	5876
Create device:							
FBA	✓	✓	✓	✓	✓	✓	✓
Celerra® FBA	✓	✓	✓	✓	✓	✓	✓
RAID-5	✓	✓	✓	✓	✓	✓	✓
RAID-6			✓	✓	✓	✓	✓
CKD	✓	✓	✓	✓	✓	✓	✓
CKD meta devices	✓	✓	✓	✓	✓	✓	✓
AS/400	✓	✓	✓	✓	✓	✓	✓
VME 512 FBA	✓	✓	✓	✓	✓	✓	✓
VDEV	✓	✓	✓	✓	✓	✓	✓
SAVE devices (for VDEVs)	✓	✓	✓	✓	✓	✓	✓
Thin devices				✓	✓	✓	✓
DATA devices				✓	✓	✓	✓
DLDEV devices					✓	✓	✓
Delete device	✓	✓	✓	✓	✓	✓	✓

**Table 2** Configuration change operations (page 2 of 4)

Configuration change operation	5671	5771	5772	5773	5874	5875	5876
Convert device:							
Adding or removing BCV/DRV	✓	✓	✓	✓	✓	✓	✓
Adding or removing RDF	✓	✓	✓	✓	✓	✓	✓
Dynamic RDF <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
Converting static SRDF to dynamic SRDF <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
Adding another mirror	✓	✓	✓	✓			
Removing a mirror	✓	✓	✓	✓			
Change device emulation:							
FBA and Celerra FBA	✓	✓	✓	✓	✓	✓	✓
Convert director type:							
FA to RDF dir/RDF to FA							✓
Virtual Provisioning:							
Create thin pools				✓	✓	✓	✓
Set pool limits				✓	✓	✓	✓
Unbind thin device from pool				✓	✓	✓	✓
Rebind thin device to pool						✓	✓
Start/stop free thin device (unwritten allocated space)				✓	✓	✓	✓
Start/stop free thin device (zero-based reclaim)					✓	✓	✓
Start/stop drain thin pool				✓	✓	✓	✓
Rebalance thin pool					✓	✓	✓
Start/stop allocate tdev				✓	✓	✓	✓
Rename thin pool						✓	✓
Thin pool compression							✓
Set device attributes:							
VCMdb	✓	✓	✓	✓			
ACLX					✓	✓	✓
RDB_Cksum	✓	✓	✓	✓			
dyn_rdf <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
dyn_rdf1_only <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
dyn_rdf2_only <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
SCSI3_persist_reserv	✓	✓	✓	✓	✓	✓	✓

**Table 2** Configuration change operations (page 3 of 4)

Configuration change operation	5671	5771	5772	5773	5874	5875	5876
RCVRPNT_TAG						✓	✓
DIF1							✓
AS400_GK							✓
Set device geometry				✓	✓	✓	✓
Set device identifiers				✓	✓	✓	✓
Map device:	✓	✓	✓	✓	✓	✓	✓
AS/400	✓	✓	✓	✓	✓	✓	✓
CKD	✓	✓	✓	✓	✓	✓	✓
Unmap device:	✓	✓	✓	✓	✓	✓	✓
AS/400	✓	✓	✓	✓	✓	✓	✓
CKD	✓	✓	✓	✓	✓	✓	✓
Manage PAV Alias	✓	✓	✓	✓	✓	✓	✓
Form meta	✓	✓	✓	✓	✓	✓	✓
Add metamember:							
Concatenated	✓	✓	✓	✓	✓	✓	✓
Striped <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
Remove metamember:							
Concatenated	✓	✓	✓	✓	✓	✓	✓
Dissolve Meta	✓	✓	✓	✓	✓	✓	✓
Swap RA group	✓	✓	✓	✓	✓	✓	✓
Swap hyper (Optimizer)	✓	✓	✓	✓			
Swap SRDF device:	✓	✓	✓	✓	✓	✓	✓
Dynamic SRDF devices <sup>a</sup>	✓	✓	✓	✓	✓	✓	✓
Set front-end port attributes	✓	✓	✓	✓	✓	✓	✓
Set Symmetrix-wide parameters:							
auto_meta				✓	✓	✓	✓
auto_meta_config				✓	✓	✓	✓
auto_meta_member_size				✓	✓	✓	✓
min_auto_meta_size				✓	✓	✓	✓
max_hypers_per_disk	✓	✓	✓	✓			
fba_multi_access_cache	✓	✓	✓	✓			
dynamic_rdf <sup>a</sup>	✓	✓	✓	✓			
raid_5_support	✓	✓					



**Table 2** Configuration change operations (page 4 of 4)

Configuration change operation	5671	5771	5772	5773	5874	5875	5876
raid_5_members	✓	✓					
concurrent_rdf <sup>a</sup>	✓	✓	✓				
dynamic_concurrent_rdf <sup>a</sup>	✓	✓	✓				
pav_alias_limit	✓	✓	✓	✓	✓	✓	✓
PAV_mode	✓	✓	✓	✓	✓	✓	✓
rdfa_cache_percent	✓	✓	✓	✓	✓	✓	✓
rdfa_host_throttle_time	✓	✓	✓	✓	✓	✓	✓
hot_swap_policy		✓	✓	✓			
replace disk							✓
SRDF operations:							
Add/remove SRDF mirror				✓	✓	✓	✓
Device pools:							
Create/delete pools	✓	✓	✓	✓	✓	✓	✓
Add/remove/enable/disable pool members	✓	✓	✓	✓	✓	✓	✓
Rename a pool						✓	✓
Federated Tiered Storage (FTS):							
Add external disk							✓
Add external disk group							✓
Remove external disk group							✓

a. Symmetrix 8000-series or higher.

## Setting Symmetrix metrics

You can set certain attributes to gain access to new Symmetrix array features using the following command:

```
set Symmetrix MetricName=MetricValue
[, MetricName=MetricValue];
```

Where:

*MetricName* — The Symmetrix metric to be set. Possible metric names are:

`auto_meta` — Enables the auto meta feature. This can be enabled only if the other auto meta parameters `min_auto_meta_size`, `auto_meta_config` and `auto_meta_member_size` are set to valid values. Possible values are `ENABLE` or `DISABLE`. This option is not applicable for CKD metadevices.

`auto_meta_config` — Specifies the default meta config when the `auto_meta` feature is enabled. Possible values are `CONCATENATED`, `STRIPED`, or `NONE`.

`auto_meta_member_size` — Specifies the default meta member size in cylinders when the `auto_meta` feature is enabled. Possible values are listed in [Table 7 on page 54](#).

`concurrent_rdf` — Enables access to the concurrent SRDF feature - multiple R2 devices for a single static R1. Possible values are `ENABLE` or `DISABLE`.

`dynamic_rdf` — Enables the creation of a set of devices that are RDF-capable, (can be dynamically assigned as RDF1 or RDF2 devices). Possible values are `ENABLE` or `DISABLE`.

`dynamic_concurrent_rdf` — Enables access to the concurrent SRDF feature (multiple R2 devices for a single R1) for dynamic SRDF devices. Possible values are `ENABLE` or `DISABLE`.

---

**Note:** Enginuity 5874 and higher does not support `concurrent_RDF`, `dynamic_RDF`, and `dynamic_concurrent_RDF`.

---

`fba_multi_access_cache` — Determines whether a read request can share cache slots in some conditions. Possible values are `ENABLE` or `DISABLE`.

`hot_swap_policy` — Indicates whether data from a failed disk should be relocated temporarily, or permanently moved to a spare disk. Possible values are `TEMPORARY` or `PERMANENT`.

---

**Note:** Enginuity 5874 and higher does not support `hot_swap_policy`.

---

`max_hypers_per_disk` — Specifies the maximum number of hypers that can be created on a physical disk. Possible values are 1 to 255, where the upper limit varies with the Enginuity level.

---

**Note:** Enginuity 5874 and higher does not support `max_hypers_per_disk`. The maximum number of hypers per physical disk is set to 512 for 5874 and 1024 for 5875.

---

`min_auto_meta_size` — Specifies the size threshold that triggers auto meta creation. When you try to create a device greater than `min_auto_meta_size` and `auto_meta` is enabled then a meta will be created. Possible values are listed in [Table 7 on page 54](#).

`pav_alias_limit` — If PAV is enabled, specifies the maximum number of aliases that can be assigned to a device. Possible values are:

- Enginuity 5671: 1-15
- Enginuity 5771/5772: 1-255

– Enginuity 5773+: 1-127

`PAV_mode` — Enables the use of PAV (Parallel Access Volumes). Possible values are:

`STANDARD` — Standard PAV volumes with static aliasing can be configured.

`DYNAMIC_STANDARD` — Standard PAV volumes with dynamic aliasing can be configured.

`rdfa_cache_percent` — Sets the percentage of write pending cache that can be used by SRDF/A. This is a value from 0 to 100 percent.

`rdfa_host_throttle_time` — Sets the number of seconds to throttle host writes to SRDF/A devices when cache is full, before dropping RDF/A sessions. Throttling delays a write from the host until a cache slot becomes free. Values are from 0 to 65535. This option is not supported on Symmetrix VMAX 10K/VMAXe arrays.

`raid_5_support` — Enables the ability to create RAID 5 devices in the Symmetrix array. Possible values are `ENABLE` or `DISABLE`. This feature is obsolete in Enginuity version 5772 and higher.

`raid_s_members` — If RAID-S support is enabled, this identifies the number of members in the Parity RAID set. Possible values are 3 or 7. This also controls the RAID 5 member count. This feature is obsolete in Enginuity version 5771 and higher.

`raid_s_support` — Enables the ability to create RAID-S or parity RAID devices in the Symmetrix array. Possible values are `ENABLE` or `DISABLE`. This feature is obsolete in Enginuity version 5772 and higher.

`VCMDB_restricted_access` — When enabled, restricts host access to the VCM database as indicated by the user. Possible values are `ENABLE` or `DISABLE`. This metric is obsolete in Enginuity version 5771 and higher.

## Viewing the Symmetrix metrics

You can view the current settings for these metrics with the following command:

```
symcfg -sid SymmID -v list
```

## Subsystem IDs for mainframe systems

On Symmetrix arrays with FICON or ESCON adapters, the `mvs_ssid` and `remote_mvs_ssid` are optional parameters when creating FBA devices through the `symconfigure` CLI. The array selects a unique `mvs_ssid` for the newly created FBA device. The parameter is optional; if an `mvs_ssid` is explicitly specified, will be honored. The `mvs_ssid` is required when a CKD device is being created.

To obtain a list of SSIDs that are currently in use, use the following form:

```
symcfg list -ssid -sid SymmID
```

## Planning system protection configurations

The SYMCLI maintains configuration and status information for every device on each of the Symmetrix arrays that are accessible from the host. A Symmetrix device can map to a part of a physical disk or to an entire disk. The part of a physical disk to which a Symmetrix device maps is called a *hypervolume*.

### Device mirrors

There are two Symmetrix device configurations that involve various mirror set protection possibilities. [Table 3](#) shows the mirror positions for the Engenuity 5874 and higher virtualized mirror positions. [Table 4 on page 37](#) provides the mirror protection scheme for arrays running Engenuity 5773 and earlier.

**Table 3** Virtual mirror positions (page 1 of 2)

Device/mirror configuration	M1	M2	M3	M4
2-Way-BCV-Mir	RAID 1			
2-Way-Mir	RAID 1			
BCV	Data			
BCV+R-5 <sup>a</sup>	RAID 5			
BCV+R-6 <sup>a</sup>	RAID 6			
BCV+TDEV <sup>b</sup>				
2-Way-DRV-Mir	RAID 1			
Parity RAID 3+1 (in multiples of 3)	RAID 5			
Parity RAID 7+1 (in multiples of 7)	RAID 5			
RAID 6	RAID 6			
RDF1	Data	Remote Data		
RDF1+R-5 <sup>a</sup>	RAID 5	Remote Data		
RDF1+R-6	RAID 6	Remote Data		
RDF1-BCV	Data	Remote Data		
RDF1-BCV+R-5 <sup>a</sup>	RAID 5	Remote Data		
RDF1-BCV+R-6	RAID 6	Remote Data		
RDF1-BCV-Mir	RAID 1	Remote Data		
RDF1-Mir	RAID 1	Remote Data		
RDF2	Remote Data	DATA		
RDF2+R-5 <sup>a</sup>	Remote Data	RAID 5		
RDF2+R-6	Remote Data	RAID 6		
RDF2-BCV	Remote Data	DATA		
RDF2-BCV+R-5 <sup>a</sup>	Remote Data	RAID 5		
RDF2-BCV+R6	Remote Data	RAID 6		

**Table 3** Virtual mirror positions (page 2 of 2)

Device/mirror configuration	M1	M2	M3	M4
RDF2-BCV-Mir	Remote Data	RAID 1		
RDF2-Mir	Remote Data	RAID 1		
RDF21	Remote Data	Remote Data	DATA	
RDF21+R-5	Remote Data	Remote Data	RAID 5	
RDF21+R-6	Remote Data	Remote Data	RAID 6	
RDF21-Mir	Remote Data	Remote Data	RAID 1	
TDEV	Thin			
RDF1+TDEV	Thin	Remote Data		
RDF2+TDEV	Remote Data	Thin		
Unprotected	DATA			
VDEV	Virtual			
RDF1 +DLDEV		Remote Data		
RDF2+DLDEV	Remote Data			
RDF21+DLDEV	Remote Data	Remote Data		

1. This table item is maintained for historical reasons. RAID 5 devices are built on either four or eight hyper members, where data and parity are striped across all members.
2. Only allowed with 5874.229.182 or higher.

[Table 4](#) lists the mirror set protection schemes for devices on Symmetrix arrays running Enginuity 5773 and earlier.

**Table 4** Mirror set protection schemes (page 1 of 2)

Device/mirror configuration	M1	M2	M3	M4
2-Way-BCV-Mir	Data	Data		
2-Way-Mir	Data	Data		
3-Way-Mir	Data	Data	Data	
4-Way-Mir	Data	Data	Data	Data
BCV	Data			
BCV+R-5 <sup>a</sup>	Data and Parity	Data and Parity		
BCV+R-6 <sup>a</sup>	Data and Parity			
DRV	Data			
2-Way-DRV-Mir	Data	Data		
Parity RAID 3+1 (in multiples of 3)	RAID Data	RAID Parity (shared)		
Parity RAID 7+1 (in multiples of 7)	RAID Data	RAID Parity (shared)		
RAID 5 <sup>a</sup>	Data and Parity	Data and Parity		
RAID 6	Data and Parity			

**Table 4** Mirror set protection schemes (page 2 of 2)

Device/mirror configuration	M1	M2	M3	M4
RDF1	Data	Remote Data		
RDF1+R-5 <sup>a</sup>	Data and Parity	Remote Data	Data and Parity	
RDF1+R-6	Data and Parity	Remote Data		
RDF1-BCV	Data	Remote Data		
RDF1-BCV+R-5 <sup>a</sup>	Data and Parity	Data and Parity		
RDF1-BCV+R-6	Data and Parity	Remote Data		
RDF1-BCV-Mir	Data	Remote Data	Data	
RDF1-Mir	Data	Remote Data	Data	
RDF1-R-S	RAID Data	Remote Data	RAID Parity	
RDF2	Remote Data	Data		
RDF2+R-5 <sup>a</sup>	Remote Data	Data and Parity	Data and Parity	
RDF2+R-6	Remote Data	Data and Parity		
RDF2-BCV	Remote Data	Data		
RDF2-BCV+R-5 <sup>a</sup>	Remote Data	Data and Parity	Data and Parity	
RDF2-BCV-Mir	Remote Data	Data	Data	
RDF2-Mir	Remote Data	Data	Data	
RDF2-R-S	Remote Data	RAID Data	RAID Parity	
RDF21	Remote Data	Remote Data	Data	
RDF21+R-5	Remote Data	Remote Data	Data and Parity	Data and Parity
RDF21+R-6	Remote Data	Remote Data	Data and Parity	
RDF21-Mir	Remote Data	Remote Data	Data	Data
TDEV <sup>b</sup>	N/A			
Unprotected	Data			
VDEV	Virtual			
DLDEV	N/A			

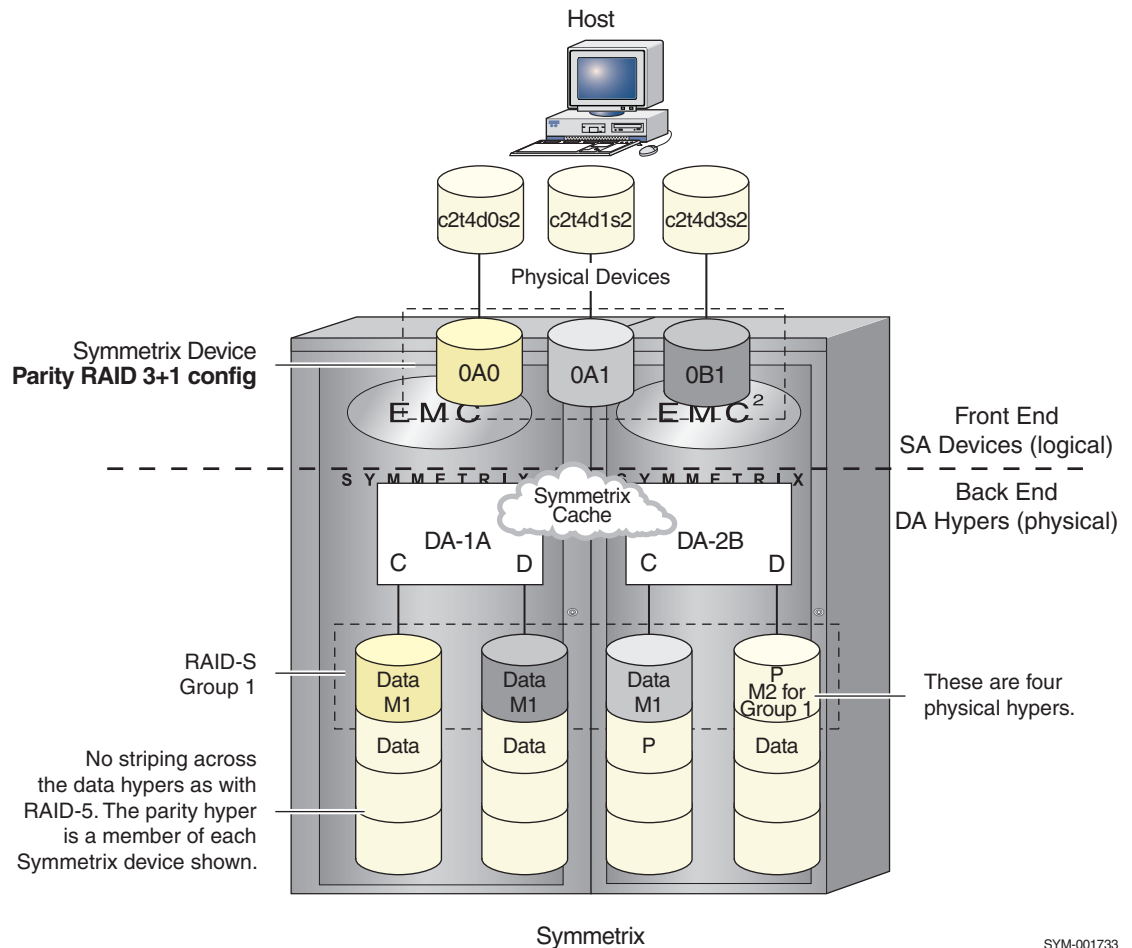
a. This table item is maintained for historical reasons. RAID 5 devices are built on either four or eight hyper members, where data and parity are striped across all members.

b. Only available with Enginuity V5773 and higher.

## Parity RAID protection

Parity RAID devices must be created in multiples of three or seven. For example, a Parity RAID 3+1 protection group requires three devices for data and one for shared parity. Any requests to create parity devices outside of the three or seven multiple factor will be denied. The Symmetrix array maps each of the device's data mirrors to the same mirror positions on a separate hyper and creates one parity device to be shared by the DATA devices. The number of Parity RAID (three or seven members) devices must be defined in the Symmetrix attributes.

Figure 1 illustrates the devices of a parity RAID mirror configuration (3+1).



**Figure 1** Symmetrix parity RAID mirror scheme

You can use the `symdev list` command to get a list of all the available devices. You can then choose a device from this list and get the configuration scheme it is serving by using the `symdev show` command. This information then leads you to back-end information for the device's disk director(s) and corresponding hyper volumes, and their mapping to disk drives. For more information, refer to the *EMC Solutions Enabler Symmetrix CLI Array Management Product Guide*.

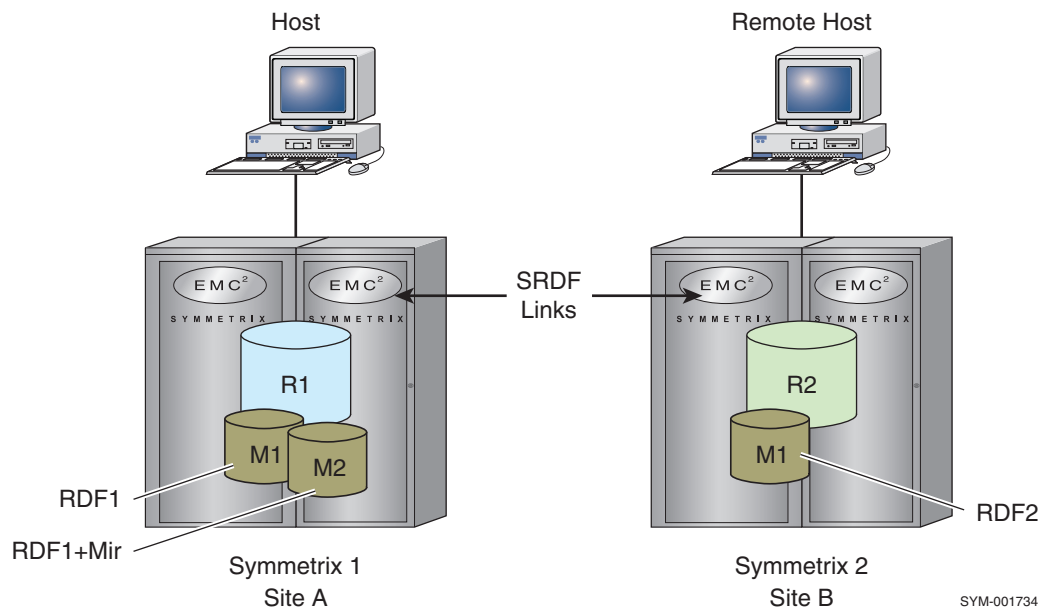
## SRDF remote protection

The Symmetrix Remote Data Facility (SRDF®) is a business continuance solution that maintains a mirror image of data at the device level in Symmetrix arrays located in physically separate sites.

## SRDF devices

SRDF configurations provide for either a unidirectional or a bidirectional data transfer over SRDF links between sites. Devices are designated as either R1 (source) or R2 (target) mirrors to synchronize and coordinate SRDF transfer activity. Devices can also be concurrent mirrors, such as an R11 device that has two separate R2 pair relationships with R2 devices in different Symmetrix arrays. In addition, a device can have a dual role of both source mirror (R1) to a target mirror (R2) and target mirror (R2) to a source device (R1). This device is called an R21. R21 devices are used in cascaded SRDF configurations. Refer to the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide* for more details about cascaded SRDF devices.

As shown in [Figure 2](#), a source R1 device is designated an RDF1 type, while a target R2 device is designated an RDF2 type. An R1 or an R2 device can also have associated mirror(s) (M1, M2) for increased protection. These are designated RDF1+Mir or RDF2+Mir as configuration types when using `symconfigure`.



**Figure 2** Simple SRDF device configuration

## SRDF and BCV devices

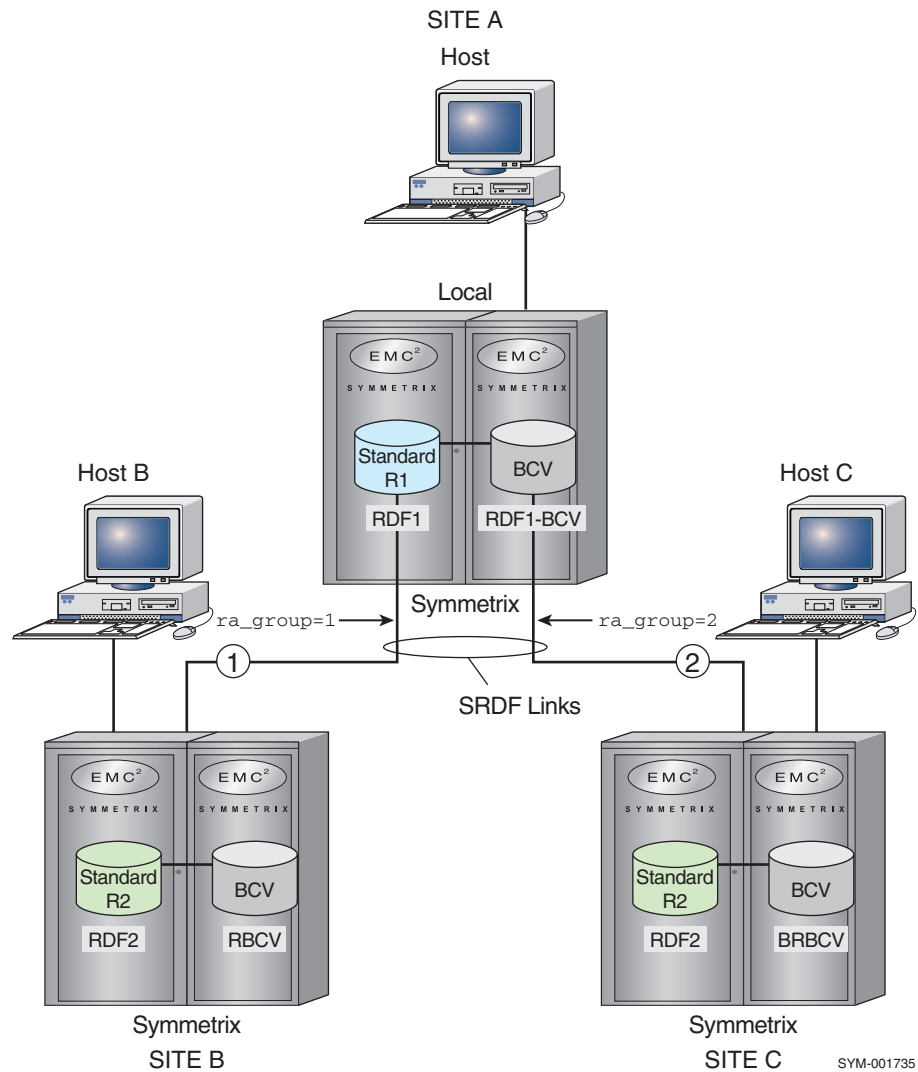
You can configure both SRDF and TimeFinder® (BCV) devices that might be in various compounded remote configurations.

As shown in [Figure 3 on page 41](#), 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 can also have a third site (remote Site C) to remotely mirror just the BCV devices in the Symmetrix at Site A.



There are two types of SRDF-connected BCV devices:

- ◆ The SRDF-connected BCVs can be paired with R2 mirrors of the local SRDF standard devices. In this case, the BCV is designated RBCV, which identifies it as remote BCV device.
- ◆ The SRDF-connected BCVs can be paired with the R2 mirrors of the local SRDF BCV devices, designated RDF1-BCV. The SRDF-connected BCV devices can be remotely associated with one device group of any type. In this case, a BCV device designated BRBCV, which identifies this device as a BCV on a remote host originating from a BCV.



**Figure 3** Various SRDF-connected device configurations with BCVs

## Managing external disk groups

Solutions Enabler V7.4 introduced a new feature called Federated Tiered Storage (FTS). With FTS, an external LUN (eDisk) can be attached through the SAN to the Symmetrix array and can be used as external back-end disks for that Symmetrix array.

Refer to [Chapter 5, “Federated Tiered Storage,”](#) for configuration details.

The following disk group management features are available:

- ◆ Create and name an empty external disk group
- ◆ Delete an empty external disk group

---

**Note:** Creating and deleting internal disk groups is not supported.

---

To use FTS, you must set up an external disk group using the `symconfigure` command. Use the following form to create an empty disk group:

```
create disk_group DskGrpName disk_location = external;
```

The disk group name serves as an additional identification mechanism and does not replace disk group numbers.

### Example

To create a disk group for a storage pool and give it the name `hr_sg_pool`, enter the following from the command line:

```
symconfigure -sid 2012 -cmd "create disk_group hr_disk_group
disk_location = external;" commit
```

---

**Note:** Refer to *EMC Solutions Enabler Symmetrix Array Management Product Guide* for all the list and show displays for the `syndisk` command.

---

## Restrictions

The following restrictions apply when creating disk groups:

- ◆ The maximum number of external disk groups that can be created on a Symmetrix is 512.
- ◆ A disk group cannot be created and deleted in the same session.
- ◆ The specified disk group name must be unique; the check for uniqueness is case insensitive.
- ◆ The specified disk group name cannot exceed 32 characters. Only alphanumeric characters, hyphens "-" and underscores "\_" are allowed. The name cannot start with a hyphen "-" or underscore "\_".
- ◆ The disk group name given cannot have the same format as the default disk group names, for example `DISK_GROUP_001`.

## Deleting a disk group

The following form will be used to delete a disk group by disk group number or by disk group name:

```
delete disk_group <DskGrpNum | name:DskGrpName>;
```

*Example* To delete a disk group with the name `hr_disk_group`, enter the following from the command line:

```
symconfigure -sid 2012  
-cmd "delete disk_group name:hr_disk_group;" commit
```

## Restrictions

The following restrictions apply when deleting disk groups:

- ◆ The disk group must be empty.
- ◆ The disk group name or number must exist.
- ◆ Only external disk groups can be deleted.

Physical disks and disk groups have some rules that need to be observed by this new feature, as follows:

- ◆ Managing external disk groups requires Enginuity release 5876 or higher.
- ◆ An external disk group can only contain external disks.
- ◆ Every external disk in the Symmetrix array must belong to an external disk group.

## Managing devices

Creation of certain device (configuration) types may require two sessions. [Table 5](#) describes the sessions required to create a device type and/or convert the various types of devices from one configuration type to another.

**Table 5** Steps to create a device (page 1 of 3)

Desired device type	Session 1 create	Session 2
2-Way-BCV-Mir	2-Way-BCV-Mir	
2-Way-Mir	2-Way-Mir	
3-Way-Mir <sup>a</sup>	3-Way-Mir	
4-Way-Mir <sup>b</sup>	4-Way-Mir	
BCV	BCV	
BCV+TDEV	BCV+TDEV	
RDF1-BCV+TDEV	BCV+TDEV	Convert to RDF1-BCV+TDEV
RDF2-BCV+TDEV	BCV+TDEV	Convert to RDF2-BCV+TDEV
DRV <sup>c</sup>	DRV	
2-Way-DRV-Mir <sup>g</sup>	2-Way-DRV-Mir	
Parity RAID	RAID-S <sup>d</sup>	
RAID 5	RAID-5	
RAID 6	RAID-6	
RAID 5 BCV	BCV+R-5	

**Table 5** Steps to create a device (page 2 of 3)

Desired device type	Session 1 create	Session 2
RAID 6 BCV	BCV+R-6	
RDF1	RDF1	
RDF1+R-5	RDF1+R-5	
RDF1+R-6	RDF1+R-6	
RDF1-BCV	BCV	Convert to RDF1-BCV
RDF1-BCV+R-5	RDF1+R-5	Convert to RDF1-BCV+R-5
RDF1-BCV+R-6	RDF1+R-6	RDF1-BCV+R-6
RDF1-BCV-Mir	RDF1+Mir	RDF1-BCV-Mir
RDF1-Mir	RDF1-Mir	
RDF2	RDF2	
RDF2+R-5	RDF2+R-5	
RDF2+R-6	RDF2+R-6	
RDF2-BCV	BCV	RDF2-BCV
RDF2-BCV+R-5	RDF2+R-5	RDF2-BCV+R-5
RDF2-BCV+R-6	RDF2+R-6	RDF2-BCV+R-6
RDF2-BCV-Mir	RDF2+Mir	RDF2-BCV-Mir
RDF2-Mir	RDF2-Mir	
RDF21 <sup>e</sup> +R5	RDF1+R5	Add R2 mirror
RDF21 <sup>e,f</sup> +R5	RDF2+R5	Add R1 mirror
RDF21 <sup>e,f</sup> +R6	RDF1+R6	Add R2 mirror
RDF21 <sup>e,f</sup> +R6	RDF2+R6	Add R1 mirror
RDF21 <sup>e,f</sup> -Mir	RDF2-Mir	Add R1 mirror
RDF22 <sup>f,g</sup> +R5	RDF2+R5	Add R2 mirror
RDF22 <sup>f,g</sup> +R6	RDF2+R6	Add R1 mirror
RDF1+TDEV	RDF1+TDEV	
RDF2+TDEV	RDF2+TDEV	
RDF1-BCV+TDEV	RDF1+TDEV	Convert to RDF1-BCV+TDEV
RDF2-BCV+TDEV	RDF2+TDEV	RDF2-BCV+TDEV
DLDEV <sup>g</sup>		
RDF1+DLDEV <sup>g</sup>		
RDF2+DLDEV <sup>g</sup>		
RDF21+DLDEV <sup>g</sup>	RDF2+DLDEV	AddR1 mirror

**Table 5** Steps to create a device (page 3 of 3)

Desired device type	Session 1 create	Session 2
RDF21+DLDEV <sup>g</sup>	RDF1+DLDEV	AddR2 mirror
Thin device (TDEV) <sup>e</sup>	TDEV	
Unprotected	Unprotected	
Virtual	VDEV	

- a. 3-Way-Mir devices are not supported on Enginuity 5773 and higher.
- b. 4-Way-Mir devices cannot have any BCV devices. 4-Way-Mir devices are not supported on Enginuity 5874 and higher.
- c. Not supported on Enginuity 5874 and higher.
- d. Not allowed for Enginuity 5771 and higher.
- e. Only for Enginuity 5773 and higher.
- f. To create an R21 or R22 device, use add rdf mirror with the symconfigure command.
- g. Only for Enginuity 5874 and higher.

For information on creating dynamic-RDF capable devices, refer to [“Configuring dynamic RDF-capable devices” on page 93](#).

When you create an SRDF device where a corresponding device must be created on a remote SRDF-linked Symmetrix, a configuration change session is initiated and managed on the remote Symmetrix by the local `symconfigure` utility.

---

**Note:** SAVE devices and DATA devices are created following the same procedure as for standard protected devices, with the additional specification of `attribute=savedev` or `attribute=datadev`.

---

## Creating devices

In a command file, you can create (add) one or more Symmetrix devices in a specified Symmetrix array.

To create one or more devices, use the following command:

```
create dev count=n,
size=n [MB |GB | CYL],
emulation=EmulationType,
config=DeviceConfig,
[, data_member_count=nn]
[, remote_config=DeviceConfig,
remote_data_member_count=nn,
ra_group=n, [remote_mvs_ssid=nnn],
[dynamic_capability=[dyn_rdf | dyn_rdf1_only | dyn_rdf2_only]]
[, mvs_ssid=nnn]
[, attribute=ckd_meta | savedev | datadev
[in pool PoolName] [member_state=ENABLE | DISABLE] ]
[meta_member_size=n [MB | GB | CYL]]
[meta_config=n [striped | concatenated]]
[, disk_group=nnn| name:DskGrpName,
[remote_disk_group=nnn | name:DskGrpName] ],
[, binding to pool=PoolName
[preallocate size=n [MB |GB | CYL]
[allocate_type = persistent]
[remote_pool=PoolName]]
[, [mapping to dir DirNum:PortNum
[starting] target = <scsi_target>,
lun=scsi_lun, vbus=fibre_vbus
[starting] base_address <cuu_address>]...]
[[device_attr =
[SCSI3_persist_reserv | ACLX |DIF1 |AS400_GK]]...];
```

Where:

**count** — Indicates the number of devices to create.

**size** — Specifies the size of the device needed in number of cylinders (default), megabytes, or gigabytes.

**Table 6** Maximum device sizes by Enginuity version

Enginuity version	MBs	CYLs	GBs
Enginuity 5874 and higher	245760	262668	240
Enginuity 5773 and lower	61425	65520	59

To calculate the number of cylinders (for pre-Symmetrix DMX™), use either of the following:

$\text{blocks} \div 960$  or  $(\text{size in megabytes}) \times 2.1333$

To calculate the number of cylinders (for Symmetrix DMX and Symmetrix VMAX Family arrays), use the following:

1 cylinder = 15 tracks; each track is 64 KB

15 x 64 tracks = .937 MB for each cylinder

---

**Note:** Devices to be used as BCV, RDF, or metamembers will need to precisely match corresponding device sizes. Use `symdev/sympd show` or `symdev/sympd list -cyl` commands to see relevant device sizes.

---

Sizing limits for CKD devices are:

CKD 3380 — 3393 cyls

CKD 3390 — 262668 cyls

For additional information about sizing devices in cylinders, refer to Chapter 3 of the *EMC Solutions Enabler Symmetrix CLI Array Management Product Guide*.

`emulation` — Specifies the device emulation type.

With Solutions Enabler and Enginuity 5875 and higher, you can create an AS/400 volume of any size and meta configuration for the following emulation types:

`AS/400_M2107_099` — Specifies a volume type 2107\_A99 to be treated as protected (equivalent to the current A0x models).

`AS/400_M2107_050` — Specifies a volume type 2107\_A50 to be treated as unprotected (equivalent to the current A8x models).

If `auto_meta` is enabled, the command follows the `auto_meta` rules for non-AS400 FBA devices to create metadevices. Also, similar to non AS/400 FBA devices, the `meta_member_size` and `meta_config` can be used to override the `auto_meta` settings on the system.

---

**Note:** The protection interpretation by the OS is not related to the actual Symmetrix RAID protection.

---

`config` — Specifies the desired device configuration type. Possible values are shown in the second column of [Table 5 on page 43](#).

`data_member_count` — Indicates the number of data members when creating a RAID 5 or RAID 6 devices on a Symmetrix array with Enginuity version 5772 and higher.

Value for <code>data_member_count</code>	Protection Type
3	RAID5 (3+1)
7	RAID5 (7+1)
6	RAID6 (6+2)
14	RAID6 (14+2)

`remote_config` — Specifies the desired remote SRDF configuration (if any) from [Table 5 on page 43](#).

`remote_data_member_count` — Specifies the number of remote data members when creating SRDF RAID 5 or RAID 6 devices on a Symmetrix array with Enginuity version 5772 and higher. You should set the value to 3 or 7 for RAID 5 (3+1) and RAID 5 (7+1) or, 6 or 14 for RAID 6 (6+2) and RAID 6 (14+2).

`ra_group` — Specifies the RA group number in the SRDF environment.

`remote_mvs_ssid` — When creating an SRDF device in a remote Symmetrix array that also contains CKD devices, a z/OS (MVS) subsystem ID (`remote_mvs_ssid`) value may be provided so the new FBA devices are not seen as part of an existing subsystem ID group. Only one `mvs_ssid` and `remote_mvs_ssid` can be used in a session. They will be applied to all devices created within that session.

`dynamic_capability` — Specifies the type of dynamic SRDF device to create. Possible values are:

- `dyn_rdf` — Creates a dynamic SRDF device.
- `dyn_rdf1_only` — Creates a dynamic R1 SRDF device.
- `dyn_rdf2_only` — Creates dynamic R2 SRDF device.

With Enginuity 5874 and higher, a single device can be set to be dynamic, without automatically creating a device pair.

**Note:** Enginuity 5874 Q22011 SR supports creating thin BCV devices (BCV+TDEV and BCV-TDEV) with the dynamic capability set (`dyn_rdf`, `dyn_rdf1_only`, and `dyn_rdf2_only`). These devices can later be converted to be thin SRDF BCV devices. Refer to [“Converting devices” on page 64](#).

When creating a dynamic pair, the specified dynamic capability is applied to the local device, and the corresponding remote device is assigned a complementary dynamic capability, according to the following:

Local device	Remote device
<code>dyn_rdf</code>	<code>dyn_rdf</code>
<code>dyn_rdf1_only</code>	<code>dyn_rdf2_only</code>
<code>dyn_rdf2_only</code>	<code>dyn_rdf1_only</code>

`mvs_ssid` — When creating a device in a Symmetrix array that also contains CKD devices, a z/OS (MVS) subsystem ID (`mvs_ssid`) value can be provided so the new FBA devices are not seen as part of an existing subsystem ID group. For more information about SSIDs, refer to [“Subsystem IDs for mainframe systems” on page 35](#).

**Note:** The `mvs_ssid` and `remote_mvs_ssid` parameters are optional when creating FBA devices on Symmetrix arrays that have FICON or ESCON adapters.

`attribute` — Specifies the desired device attribute. Possible values are:

`ckd_meta` — When creating a device with emulation type of CKD-3390, this indicates that the device should be a striped metadvice. CKD metadevices must be created in sets of four devices.

`savedev` — When creating a device, this indicates that the device should be a SAVE device. The device will become part of a pool of devices that are used with TimeFinder/Snap for virtual device Snap operations.

`datadev` — When creating a device, this indicates that the device should be a DATA device. The device will become part of a pool of devices that are used with thin devices for virtual operations.



---

**Note:** If the Symmetrix array is running Enginuity 5874 and Solutions Enabler V7.1 or higher, a disk group must be specified when creating a SAVE device or DATA device if the `create dev` command is also adding the device(s) to a pool.

---

*PoolName* — Specifies the name of the SAVE or thin pool. It can be from 1 to 12 alphanumeric characters long and include hyphens (-), and underscore (\_) characters. The name `DEFAULT_POOL` is reserved to represent the container of all *unpooled* SAVE devices. Thin pools do not use the `DEFAULT_POOL`.

---

**Note:** The pool name `DF_DDEV_POOL` is reserved for EMC use only.

---

*member\_state* — Indicates whether the device(s) being added should be enabled or disabled in the pool.

---

**Note:** Refer to [“Managing metadevices” on page 53](#) for details about creating metadevices using the `meta_member_size` and `meta_config` options.

---

*disk\_group* — Specifies a disk group number (or name) to which the created device belongs. A disk group is a set of physical disks set aside to be used to create devices of a specific protection level. Disk group usage may improve the performance of some configurations.

*name* — Specifies the name of the disk group. By default, the disk group name is `DISK_GROUP_xxx`, where `xxx` is the disk group number.

Usage is: `disk_group=#` or `disk_group=name:DskGrpName`

*remote\_disk\_group* — Specifies a remote disk group number (or name) when creating a device.

*name* — Specifies the name of the remote disk group. By default, the disk group name is `DISK_GROUP_xxx`, where `xxx` is the disk group number.

*binding to pool* — Specifies the thin *PoolName*. This option is only available when creating thin devices. When creating SRDF thin devices, both SRDF devices must be bound at the same time. Individual RDF1 and RDF2 devices cannot be bound separately. Devices are not required to be bound to a pool upon creation, however, they cannot be used until they are bound to a pool.

*preallocate size* — Indicates the amount of space (pre) allocated to the thin device(s) when it is bound to a pool. If MB or GB is not specified, the size defaults to cylinders.

*allocate\_type* — An optional parameter that can be specified along with the `preallocate_size` option. With this option, the allocations are unaffected by any reclaim operations, as well as clone, snap, or SRDF copy operations. If it is not supplied, `symconfigure` will preallocate non persistent tracks which can be reclaimed without any additional reclaim flags.

*remote\_pool* — Specifies the name of the remote thin pool. This option is only for SRDF thin devices.

*mapping to dir* — Specifies the director and port for the mapping operation.

*scsi\_target* — is a hex value for the SCSI target ID.

`scsi_lun` — Indicates a hex value for the SCSI logical unit number.

`fibre_vbus` — The virtual bus address used when mapping to an FA port using volume set addressing.

`cuu_address` — Indicates a base or alias address for a device being mapped to an EA or EF port. These are mainframe ports which expect devices to be mapped in groups to form CU images.

`device_attr` — Specifies the attributes to be set on the new device. Possible values include:

- `RDB_Cksum` (Enginuity 5773 and earlier)
- `VCMdb` (for device masking on Enginuity 5773 and earlier)
- `SCSI3_persist_reserv` (persistent group reservation) In Enginuity 5874 and lower, this attribute is disabled by default. In Enginuity 5875 and higher, this attribute is enabled by default. When running Solutions Enabler V7.5 with Enginuity 5876 Q42012SR and higher, you can set or clear the SCSI-3 persistence attribute only if the device is unmapped.
- `dyn_rdf` (This option provides the most flexibility in performing dynamic SRDF operations)
- `dyn_rdf1_only`
- `dyn_rdf2_only`
- `ACLX` (for Auto-Provisioning Groups on Enginuity 5874 and higher)
- `RCVRPNT_TAG` (for RecoverPoint with Enginuity 5875 Q2 2011 SR and higher)
- `DIF1` (Enginuity 5876 and higher)
- `AS400_GK` (Enginuity 5876 and higher)

---

**Note:** Refer to [“Setting device attributes” on page 70](#) for a list of the device attribute restrictions.

---

The output displayed during the config change session shows the types of devices created.

*Example* The output from creating devices from the device file `create_vp`, shows the device types as follows:

```
symconfigure -sid 266 -f create_vp -v -noprompt preview
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000192600266
{
  create dev count=3, size=2000 cyl, emulation=FBA, config=2-Way Mir,
    mvs_ssid=0, disk_group=5, attribute=datadev;
  create dev count=1, size=2000 cyl, emulation=FBA, config=TDEV,
    mvs_ssid=0;
  create dev count=1, size=1000 cyl, emulation=FBA, config=TDEV,
    mvs_ssid=0;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Validating configuration changes.....Validated.

New symdevs: 06BA:0CBC [DATA devices]
New symdevs: 06BD:06BE [Thin Concatenated meta, head 06BD, member size
1500 cyl]
```

```
New symdev: 06BF [TDEV]
```

```
Closing configuration change request.....Closed.
Terminating the configuration change session.....Done.
```

```
The configuration change session has completed successfully.
```

## Restrictions on creating devices

There are restrictions on the use of unprotected devices created by `symconfigure`, or converted to unprotected devices. Such devices cannot be mapped to hosts for normal data storage use. If an unprotected device is intended for use as a gatekeeper, it should be created with 48 cylinders or less in order to be mapped to a host.

You can create devices with a variety of emulation types: Fixed Block Architecture (FBA), Celerra FBA or VME512 FBA, and CKD for z/OS environments.

You cannot create or convert to SRDF device types when:

- ◆ Domino mode is enabled on any current SRDF pairs.
- ◆ There are any invalid tracks on any of the current SRDF devices.
- ◆ Concurrent SRDF is enabled on a device.
- ◆ For AS/400 types, 2107\_A99 and 2107\_A50, a size must be specified. The size range allowed for these devices is a minimum of 165 cylinders and a maximum of 2236963 cylinders.
- ◆ An AS/400 device or a device tagged for RecoverPoint cannot be mapped to an FCoE director.

When creating devices in an z/OS environment, all new devices must have the same MVS SSID and any remote devices must also have the same remote MVS SSID within a single session. To create devices with different SSID values, you will need to run more than one session.

## SRDF device pairs

With Solutions Enabler V7.5 with Enginuity 5876 Q42012SR, when you create an RDF device pair and do not provide the `mapping to dir` option, the devices are RW on the link and are marked synchronized immediately. This means that no data copy occurs. The devices immediately report a Synchronized RDF pair state.

## SRDF mode default

The SRDF mode on an SRDF device pairing will be Adaptive Copy Disk by default, unless the `SYMAPI_DEFAULT_RDF_MODE` is set in the option file. If the device being created is a diskless R1 device, the RDF mode will default to Adaptive Copy Write Pending, regardless of the option file setting. (Applies to Symmetrix arrays running Enginuity 5876 and higher.)

Symmetrix arrays running Solutions Enabler versions lower than V7.4 continue to set the SRDF mode to synchronous by default.

## Gatekeeper devices

Use the `symconfigure create gatekeeper` command to create gatekeeper devices. All information about gatekeepers can be found in the *Solutions Enabler Installation Guide* and the EMC Knowledgebase solution emc255976 available on the EMC online support web site.

## VDEV configuration restrictions

Symmetrix virtual devices are configured as full-size devices, but use minimal disk storage on the back-end. On Symmetrix DMX arrays, VDEVs reside in cache only and are depicted as having no back-end hypers. The format must be FBA. Virtual devices can also be used to form a VDEV metadvice. Other Symmetrix devices cannot be converted to form virtual devices. To configure a virtual device, set `config=vdev` in the device file.

## SAVE device configuration restrictions

Symmetrix SAVE devices work in conjunction with virtual devices and can be unprotected<sup>1</sup>, mirrored, or RAID 5/RAID 6.<sup>2</sup> SAVE devices cannot be part of a metavolume or grouped. SAVE devices cannot be converted to other Symmetrix devices. Configuring virtual and SAVE devices requires careful planning to anticipate what percentage of the data may change during a virtual copy session. Creating the SAVE device pool too small could result in data changes being lost during a copy session. Make sure that you create enough SAVE devices and set the size of the SAVE devices large enough to handle any expected changes. To configure a SAVE device, set `attribute=savedev` in the request to create a new device.

## Copying a device

You can configure available disk space into a set of new devices by copying the attributes of an existing device. To do this, either specify the quantity of disk space to configure into the new devices, or the number of devices to be created. The quantity of disk space represents the new space that will be available for the host to use, and not the space allocated in the Symmetrix array to manage the request according to the device protection requirements. In addition, you can also override some of the copied device's attributes in the new devices. For example, you can copy the attributes of a standard device, but configure the new devices as BCVs.

To configure a device by copying a similar device, use the following form:

```
configure [n.nn [MB | GB] | nn devices]
copying dev SymDevName
[mapping to dir DirectorNum:PortNum
 [masking hba [awwn=awwn | wwn=wwn |
               iscsi=iscsi | aiscsi=aiscsi]
          host_lun=lun |dynamic_lun]] ]
[overriding
 [size=nnn [MB | GB | CYL]]
 [emulation=EmulationType]
 [config=DeviceConfig]
 data_member_count=n
 [mvs_ssid=nn]
 [disk_group=n| name:DskGrpName]];
```

Where:

`n.nn [MB | GB]` — Specifies the quantity of disk space to configure.

`nn devices` — Specifies the number of devices to create.

`SymDevName` — The Symmetrix device name of the model device.

- 
1. Configuring SAVE devices as unprotected is not recommended and requires special approval by EMC.
  2. For a given device emulation type, all SAVE devices must have the same protection.

*DirectorNum:PortNum* — The mapping attributes of the device are not copied. If the new devices are to be mapped, you must specify the director/port addresses.

*masking hba* — The masking attributes of the model device are not copied. If the new devices are to be masked, you must specify the host HBA to which the devices should be masked. This option is not available on arrays running Enginuity 5874.

*host\_lun* — Specifies the LUN addresses to be used for each device that is to be added for the host HBA.

*dynamic\_lun* — Specifies to use the dynamic LUN addressing features but does not require a LUN address for each device. The LUN addresses are assigned based on what may already be in use for that host HBA.

*overriding* — Indicates that you will be overriding some of the characteristics of the copied device.

*size* — Specifies the size of the new devices.

*emulation* — Indicates the emulation type of the new devices.

*config* — Specifies the configuration type of the new devices. Possible values are shown in the second column of [Table 5 on page 43](#).

*data\_member\_count* — Specifies the number of data members when creating a RAID 5 or RAID 6 devices on a Symmetrix array with Enginuity version 5772 and higher. You should set the value to 3 or 7 for RAID 5 (3+1) and RAID 5 (7+1) or, 6 or 14 for RAID 6 (6+2) and RAID 6 (14+2).

*mvs\_ssid* — When creating devices in a Symmetrix array that also contains CKD devices, a z/OS (MVS) subsystem ID (*mvs\_ssid*) value can be provided so the new FBA devices are not seen as part of an existing subsystem ID group.

*disk\_group* — Specifies the disk group number (or name) in which to place the new devices. If this option is not specified, the model device's disk group is *not* used. Instead, the system will place the hypers on any available disk.

*name* — Specifies the name of the disk group. By default, the disk group name is *DISK\_GROUP\_xxx*, where *xxx* is the disk group number.

Usage is: *disk\_group=#* or *disk\_group=name:DskGrpName*

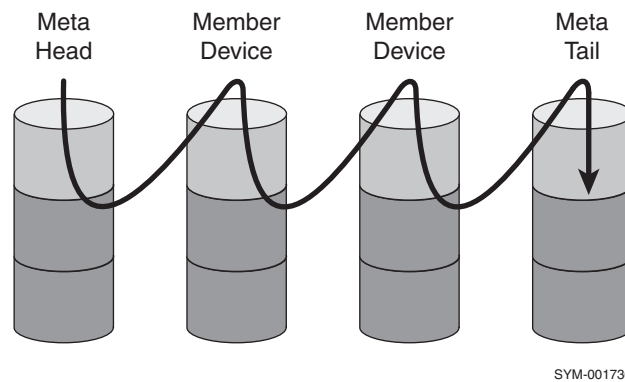
## Managing metadevices

*Metadevices* allow individual devices to be concatenated to create larger devices. The devices assigned in a meta sequence do not need to be adjacent.

The *metahead* is the first device in the metadevice and is responsible for receiving incoming commands. When an incoming command for the metahead is processed, the Enginuity software determines which metadevice member should execute the command.

Several operating systems, such as Windows and some open systems environments, can have larger volumes than are provided by standard Symmetrix physical disk devices. A metadevice is a logical volume set created from individual Symmetrix hypers to define volumes larger than the current Symmetrix maximum hyper volume size of approximately 16 GB. Metadevices are functionally the same as logical volume sets implemented with

the host volume manager software. Physically, a metadevice is two or more Symmetrix volumes presented to a host as a single addressable device. It consists of a metahead device, some number of member devices (optional), and a metatail device (see [Figure 4](#)).



**Figure 4** Concatenated metadevices

## Metadevice size requirements

Metadevices may contain as many as 255 members. The maximum size metadevice that has been tested by EMC is 60 terabytes, which is the maximum size supported. Metadevices can consist of nonsequential and nonadjacent volumes.

**Table 7** Metadevice sizes by Enginuity version for auto\_meta

Enginuity version	Max single device size (CYL)	Max single device size (GB)	Min_auto_meta_size	Auto_meta_member_size
Enginuity 5874 and higher	262668	240	262669	262668
Enginuity 5773	65520	59	65521	65520

## Meta performance

By allowing individual physical disk devices to be grouped together into a metadevice and the capability of metadevice addressing, Symmetrix enhances disk system functionality. To increase throughput and further improve performance, Symmetrix provides multiple I/O drive queues for metavolumes.

## Accessing data in a metadevice

Addressing of data contained in a metadevice can be organized in two different ways:

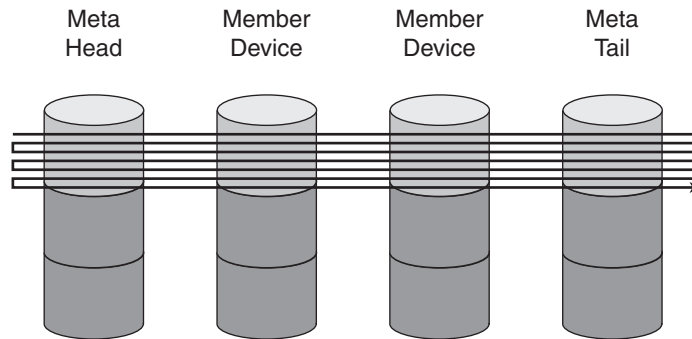
- ◆ Concatenated devices
- ◆ Striped devices

### Concatenated devices

Concatenated devices are volume sets that are organized with the first byte of data at the beginning of the first device ([Figure 4 on page 54](#)). Addressing continues to the end of the first device before any data on the next device is referenced. When writing to a concatenated device, the first metadevice member receives all the data until it is full, and then data is directed to the next member and so on.

## Striped devices

Metadevice addressing by striping divides each metamember device into a series of stripes, addressing a stripe from each device before advancing to the next stripe on the first device (as shown in [Figure 5](#)). When writing to a striped volume, equal size stripes of data from each participating drive are written alternately to each member of the set.



**Figure 5** Striped data

Striping data across the multiple drives in definable cylinder stripes benefits *sequential writes* by avoiding stacking multiple writes to a single spindle and disk director. This scheme creates a large volume, but additionally balances the I/O activity between the disk devices and the Symmetrix disk directors.

If no stripe size is specified when creating a striped meta, all Enginuity codes will consider the default stripe size as 1920 blocks of 512 bytes each:

- ◆ 1 cylinder in Enginuity 5671 and earlier = 960 512-byte blocks
- ◆ 1 cylinder in Enginuity 5771 and higher = 1920 512-byte blocks

---

**Note:** The sum of the metamember's individual cylinder size and the metahead cylinder size, may not match. A `symdev show` of the metahead device only shows the usable cylinder size.

---

## Creating metadevices

Metadevices can be created in either of the following ways:

- ◆ Automatically created (recommended) See [“Automatic metadevice creation” on page 56](#) for setting the Symmetrix array-wide meta options and [“Creating devices” on page 46](#).
- ◆ [“Forming metadevices” on page 57](#)

## Automatic metadvice creation

Symmetrix-wide meta settings can be set using the `symconfigure` command and specifying a command file. The command file supports the following meta metrics in the `set symmetrix` command:

```
set symmetrix [auto_meta = <ENABLE|DISABLE>]
               [min_auto_meta_size = n [MB | GB | CYL]]
               [auto_meta_member_size = n [MB | GB | CYL]]
               [auto_meta_config = [striped | concatenated | NONE]];
```

- ◆ `auto_meta` — This is a Symmetrix-wide setting (ENABLE/DISABLE) that can be used to enable automatic creation of metadvice members. The default value is `DISABLE`. This option is not applicable for CKD metadvice members.
- ◆ `auto_meta_member_size` — When `auto_meta` is enabled, this setting indicates the number of cylinders of each metadvice member: The default value is 0, therefore this value must be changed.
- ◆ `auto_meta_config` (striped or concatenated) — When `auto_meta` is enabled, this setting indicates the configuration can be set to striped or concatenated. The default value is `NONE`, therefore this value must be changed.
- ◆ `min_auto_meta_size` — This setting is a size threshold that triggers auto meta creation. When you try to create a device with size greater than or equal to `min_auto_meta_size`, a metadvice member will be created. Refer to [Table 7 on page 54](#) for metadvice member sizes.

For more information about setting Symmetrix array metrics, refer to [“Setting Symmetrix metrics” on page 33](#).

---

**Note:** Device reservations are honored only when devices are explicitly specified during Solutions Enabler configuration change operations. Operations that allow the Enginuity operating environment to choose devices (such as when a meta is formed and only the metahead is specified) do not honor device reservations.

---

## Restrictions

- ◆ The `min_auto_meta_size` cannot be set less than the `auto_meta_member_size`. Refer to [Table 7 on page 54](#) for metadvice member sizes.
- ◆ The metric `auto_meta` can be enabled only if `min_auto_meta_size` and `auto_meta_member_size` and `auto_meta_config` are set to valid values.
- ◆ You can override the `auto_meta_member_size` and the `auto_meta_config` in the create device command line only if `auto_meta` is `ENABLED` and the total device size is greater than `min_auto_meta_size`.
- ◆ Internal device types DATA, SAVE, and DRVs cannot be metadvice members, so the parameters `min_auto_meta_size` and `meta_member_size` are not applicable when creating these devices.
- ◆ To reset the current default `meta_member_size`, set the value to 0.



---

**Note:** The `meta_member_size` must be specified in cylinders; Solutions Enabler converts values entered with MB and GB to cylinders. The conversion formula will calculate a number "*n*" where *n* is the minimum number of cylinders required to fit the requested capacity.

---

## Forming metadevices

The automatic creation of metadevices can be accomplished with one configuration change session. Metadevice configurations that require two sessions (`form` and `convert`) are shown in [Table 8](#).

**Table 8** Steps to create an SRDF BCV metadevice

Desired device type	Session 1 form	Session 2 convert
Meta RDF1-BCV	Meta-BCV	→ Meta RDF1-BCV
Meta RDF2-BCV	Meta-BCV	→ Meta RDF2-BCV

If you do not use the automatic meta creation feature, you can form a metadevice using the `form meta` command.

When you initially create a metadevice, you must form the head and add the members in the same command file. If using the `count` option, no members need to be specified. The configuration server will automatically select members from the pool of unmapped devices that match the metahead in size, emulation, and configuration type.

---

**Note:** Make sure none of the devices being considered for a metadevice have the SCSI-3 persistent reserve flag enabled.

---

To form a metadevice, use the following command:

```
form meta from dev SymDevName,
config=MetaOption
[, stripe_size=<MetaStripeSize>[cyl]]
[, count=<member_count>;
```

where:

`config` — Specifies the meta configuration type. Possible values are `concatenated` or `striped`.

`stripe_size` — Specifies the size of the striped metadevice. This value can be expressed in blocks or cylinders. Possible sizes in 512 byte blocks are 1920, 3840, 7680, 15360, 30720, and 61440. The stripe size must be 1920, which is the default for all versions of Enginuity. If no stripe size is specified when creating a striped meta, all Enginuity codes will consider the default stripe size as 1920 blocks of 512 bytes each. For information on metavolume stripe sizes, refer to [“Striped devices” on page 55](#).

`cyl` — Indicates that the `stripe_size` is expressed in cylinders. The size of a cylinder for FBA emulation is 960 512-byte blocks.

`count` — Indicates the total number of devices for the configuration server to add to the new metadvice, including the head. This option may be omitted if you want to specify the members using the `add dev` command.

## RDF metadvicees

The `create dev` command syntax also can create static and dynamic SRDF metadvicees. To use this feature, both the local and the remote Symmetrix arrays must be running Enginuity V5773 or higher and the local Symmetrix array must have `auto_meta` enabled. The `auto_meta` feature may or may not be enabled on the remote Symmetrix array. However, any `auto_meta` settings on the remote Symmetrix array are ignored.

*Example* Assume that the local array is enabled for `auto_meta` and `min_auto_meta_size` is set to 30000. The following command file will create a 2-member metadvice, on both the local and the remote Symmetrix arrays:

```
create dev count=1, size = 60000, emulation=FBA, config=RDF1,ra_group=20,
remote_config=RDF2,dynamic_capability=dyn_rdf;
```

Note that SRDF devices cannot be used to form metadvicees in one step. When this is necessary, the SRDF devices must first be converted to non-RDF devices (as shown in [Table 8](#)), the metadvicees must be formed separately on both Symmetrix arrays, and then the metahead on one Symmetrix array can be converted back to an SRDF device.

Also, once the SRDF metadvice has been established, it cannot be modified in one step. If any modifications are required, it will again have to be converted to a non SRDF device first.

## AS/400 metadvicees

Many AS/400 devices are a fixed size. The emulation type determines the size of the device. Solutions Enabler determines the size automatically from the emulation type.

When `auto_meta` is enabled, Solutions Enabler attempts to create an AS/400 non metadvice and an AS/400 metadvice with or without a size value entered.

When `auto_meta` is disabled, Solutions Enabler will attempt to create an AS/400 non metadvice, but will reject a request to create an AS/400 metadvice.

*Example* The following command creates an AS/400 metadvice on a Symmetrix DMX array with emulation AS/400\_M4328\_50.

```
create dev count=1, emulation= AS/400_M4328_50, config=2-Way-Mir;
```

This will create an 8-member metadvice where total cylinder size is 143576 (native mode).

## Meta configuration restrictions

Solutions Enabler V7.2 and higher, with Enginuity 5875 and higher, no longer requires that non metadvicees be unmapped while being converted to a concatenated metadvice. A device being converted to a concatenated metadvice can be mapped and online to the host during the operation.

---

**Note:** A non metadvice being converted to a striped metadvice must still be unmapped before it can be converted.

---

The following are restrictions when configuring metadevices:

- ◆ Metamembers cannot be removed from a striped metadvice.
- ◆ To create a striped metadvice, all members must be the same size and have the same mirror protection.
- ◆ To remove a member from a concatenated metadvice, the member must be the tail. You cannot remove inner members.
- ◆ Only the head of a meta may have its type converted.
- ◆ Only the head of a meta is mapped or assigned to the host.
- ◆ All metas must contain at least one metamember. When a meta is formed, at least one member must be added.
- ◆ All metadevices must be composed of devices of the same BCV type (you must convert from SRDF devices in order to change metas).
- ◆ All metamembers must belong to the same physical disk group.
- ◆ Metadevices must be composed of devices that are all of the same FBA emulation type.
- ◆ You must create CKD metadevices using the `create dev` command, not the `form dev` command.
- ◆ VDEV metadevices must be composed of virtual devices that are of the same capacity, stripe size, count, and emulation type as the standard devices with which they will be used.
- ◆ An SRDF device cannot be used to form a metadvice (refer to [Table 8 on page 57](#) for converting a metadvice for SRDF operation). The configuration manager cannot form a metadvice from SRDF devices that are RDFA-protected (that is, you cannot convert these devices to non-RDF, form the metadvice, and convert them back to RDF).

## Adding members to a concatenated meta

To add additional members to an existing concatenated metadvice, use the following form:

```
add dev SymDevName[:SymDevName] to meta SymDevName;
```

*Example* To add Symmetrix devices 0013 and 0014 to concatenated meta 0010, enter:

```
add dev 0013:0014 to meta 0010;
```

The `-protect_data` option is not supported for concatenated metadevices.

## Adding members to a striped meta

Solutions Enabler supports expanding a striped thin metadvice with data protection. To add additional members to an existing striped metadvice, use the following form:

```
add dev SymDevName[:SymDevName] to meta SymDevName
[, protect_data=[TRUE | FALSE],
bcv_meta_head=SymDevName];
```

Where:

`protect_data` — Possible values are `TRUE` or `FALSE`. The `protect_data` option is only for striped metas. When set to `TRUE`, the configuration manager automatically creates a protective copy to the BCV meta of the original device striping. Because this occurs automatically, there is no need to perform a BCV establish. When set to `FALSE`, the data will not be preserved during the meta expansion.

---

**Note:** For arrays running Enginuity V5773 and earlier, RAID 5 BCVs are not supported for the `protect_data` option. You can protect data by using a mirrored BCV device.

---

`bcv_meta_head` — When adding new members to an existing, striped metadvice, if the data on the metadvice is to be protected, you must specify the name of a `bcv_meta_head` that matches the original metadvice in capacity, stripe count, and stripe size.

---

**Note:** Thin BCV devices can be used as the `bcv_meta_head`. The thin BCV metahead and all members must be bound to a thin pool.

---

*Example* To add Symmetrix devices 0013 and 0014 to striped meta 0010, enter:

```
add dev 0013:0014 to meta 0010,
protect_data=TRUE,
bcv_meta_head=00CA;
```

## Removing metamembers

To remove a metamember or members from the tail of a concatenated metadvice, use the following form:

```
remove dev SymDevName[:SymDevName] from meta SymDevName;
```

*Example* To remove Symmetrix device 0014 from meta 0010, enter the following in a command file:

```
remove dev 0014 from meta 0010;
```

## Dissolving a metadvice

The `symconfigure` command file can use a list of device numbers or ranges for the `dissolve meta` command. This allows multiple metadevices to be dissolved with one command.

To dissolve multiple metas, provide a list with the device numbers of the metahead devices, or use the device numbers of the meta heads within a device range. The command will skip over the device numbers of metamembers found in the range, dissolving only metas for the metaheads found within the range. If non metadevices are found within the range an error will return.

To dissolve one or more metadevices, use the following form:

```
dissolve meta dev SymDevName [:SymDevName]
[ [, SymDevName[:SymDevName] ] ... ];
```

**Example** To dissolve multiple metadevices where the device numbers for the metas are contiguous from `d0` to `df`, including all the metaheads, create a command file containing the following command:

```
dissolve meta dev d0:df;
```

To dissolve the meta devices, use the following command:

```
symconfigure -sid 266 commit -file dissolve_meta.cmd -v -noprompt
```

---

**Note:** A `dissolve` action removes the metahead and all its members.

---

## Data preservation concerns

When managing metadevices, data is not written directly to the devices involved, and therefore the original data is essentially preserved. However, due to the reorganization upon completion of the device management actions, the host is unable to access all of the original data. Therefore, it is the responsibility of the attached host to handle the change to the physical geometry of the device.

---

**Note:** The expansion of striped metadevices is an exception to this rule. In this case, you can provide a BCV meta to preserve the original data, which will be rewritten to the newly configured striped metadevice.

---

[Table 9](#) lists the possible data management operations and their effects on data integrity.

**Table 9** Metadevice data preservation

Metadevice type with data	Operation	Data integrity preserved status
Metadevices concatenated	Adding a member	Preserved
Metadevices concatenated	Removing a member	Preserved <sup>a</sup>
Metadevices concatenated	Dissolving to non meta	<b>Not preserved</b> <sup>b</sup>
Metadevices concatenated	Converting to striped	Configurable <sup>c</sup>
Metadevices striped	Adding a member	Configurable <sup>c</sup>
Metadevice striped	Dissolve	<b>Not preserved</b> <sup>b</sup>

a. Preserved up to where the metadevice is removed.

b. Preserved but there is no host component to piece the data together.

c. Protection is a user-configurable option.

### CAUTION

**Dissolving striped or concatenated devices cause data access loss.**

**Adding to concatenated metadevices preserves the metadevice data, but makes the new members's original data inaccessible.**

**Removing concatenated members cause data access loss on the removed member. Use these commands with great caution.**

---

## Managing spare disks

---

**Note:** Enginuity 5772 and higher does not support adding or removing spare disks.

---

With Enginuity 5771 and lower, you can reserve physical disks as dynamic (hot) spares. When a physical disk is reserved as a dynamic spare it is not allocated to any device and is held in reserve to support the hypervisors of a Symmetrix disk that has failed.

### Adding a spare disk

A spare disk can only be created from any disk containing no hypervisors. To reserve an unused disk as a spare disk, use the following command:

```
create spare count=n[, format = [512 | 520]];
```

Where:

`count` — The number (positive integer) of disks to be set aside as spares.

`format` — The recording format to be used on a spare disk. Values are 512 or 520.

### Deleting a spare disk

To remove a spare disk, making it available for normal storage, use the following command:

```
delete spare disk=[director_num, da_interface, scsi_id];
```

Where:

`director_num` — The director identity number, such as 16A.

`da_interface` — The DA SCSI path (c, d, e, or f).

`scsi_id` — The SCSI target ID (hex value).

For example:

```
delete spare disk=[16A, D, 0];
```

### Viewing existing spares

To view the set of existing spares, use the following command:

```
symdisk list -spare -v
```

The following are restrictions/conditions that can impact I/O activity:

- ◆ When adding or removing Symmetrix spares, no restriction on I/O.
- ◆ A spare cannot be removed while in use.

## Configuring thin metadevices

The same commands (`create dev`, `form meta`, `dissolve meta`, `convert meta`) can be used to configure a thin metadevice.

Follow these guidelines when creating a thin metadevice:

- ◆ On Symmetrix arrays running Enginuity versions lower than 5875, only unmapped thin devices can be formed into metadevices.
- ◆ On Symmetrix arrays running Enginuity 5875 or higher:
  - Bound thin devices can be used as metaheads; however, bound thin devices cannot be used as metamembers.
  - Unmapped thin devices can be formed into striped metadevices.
  - Mapped or unmapped thin devices can be formed into concatenated metadevices.
- ◆ Only thin devices can be used to create a thin metadevice. Mixing of thin and non thin devices to create a meta is not allowed.
- ◆ For a striped metadevice, all members should have the same device size.

### Dissolving a thin meta

Only unbound thin metadevices can be dissolved.

### Expanding a thin metadevice

The process for expanding a thin metadevice is described in [“Adding members to a concatenated meta” on page 59](#) and [“Adding members to a striped meta” on page 59](#).

Solutions Enabler with Enginuity 5875.198.148 and higher, supports the use of a thin BCV metavolume while expanding striped thin metavolumes. The expansion guidelines are:

- ◆ All existing rules for using regular BCV devices apply to thin BCV (such as size, number of members, and member size).
- ◆ A thin BCV device used for data protection must be bound to a thin pool.
- ◆ A thin BCV device used for data protection must be fully allocated with the `allocate_type=persistent` option.
- ◆ A thin BCV device cannot be used while expanding a standard striped metadevice.
- ◆ Use of thin SRDF BCVs for data protection is not allowed.

The `protect_data` option is not applicable to unbound thin devices.

A thin BCV device used for data protection must be fully allocated with the persistent option specified (explained in [“Creating thin devices \(TDEVs\)” on page 113](#)).

As a part of the expansion process if the original metadevice was bound to a thin pool, the new metamember will be bound to the same pool.

### Binding a thin metadevice to a pool

When binding a thin metadevice to a pool, use the `symconfigure` command to bind the metahead and give one preallocated size. Solutions Enabler will distribute the preallocated size among all the metamembers.

The distribution will depend on the type of the metadvice, as follows:

**Striped metadvice** — The preallocated size will be distributed evenly among the members.

**Concatenated metadvice** — Starting from the metahead, each member receives a minimum of the total preallocated size remaining, based on the specified `size_of_member`. For example, if 175 GB were preallocated for a metadvice with 3 members, each of 100 GB, Solutions Enabler will allocate 100 GB to the head and 75 GB to the 2nd member and 0 GB to the last member.

Refer to [“Configuring Virtual Provisioning” on page 109](#) for more information about binding thin devices.

## Converting devices

In the command file, you can convert the configuration type of an existing device or devices using the following command:

```
convert dev SymDevName[:SymDevName] to DeviceConfig
[emulation=EmulationType,]
[ra_group=n, remote_dev=SymDevName,
invalidate=invalidate_opt,
[remote_mvs_ssid=nnn],
start_copy=YES|NO ],
[mvs_ssid=nnn] [raidset = TRUE | FALSE];
```

Where:

*SymDevName* — Specifies the Symmetrix device name of the device targeted for change. To target more than one device, indicate the first and last devices in a series separated by a colon (:).

*DeviceConfig* — Specifies the desired device configuration type. Possible values are shown in the first column of [Table 5 on page 43](#).

*emulation* — Indicates the device’s emulation type.

*ra\_group* — Specifies the RA group number in the SRDF environment.

*remote\_dev* — Specifies the remote Symmetrix device name of the particular device targeted for change. If you specify a range of *SymDevNames* in the first line of the convert statement, the remote *SymDevName* value will be increased incrementally to arrive at the corresponding device number.

*invalidate\_opt* — Indicates the SRDF device to invalidate so that a full copy can be initiated from the remote mirror. Allowed values are R1 (invalidate the source), or R2 (invalidate the target). The value NONE is not supported in Solutions Enabler V7.0 and higher.

*remote\_mvs\_ssid* — Specifies the remote z/OS (MVS) subsystem ID that will be assigned to any device created as a result of removing any mirror(s). If not provided, the original MVS SSID will be assigned when available. If the MVS SSID group is full, you must supply a new MVS SSID. Only one *remote\_mvs\_ssid* can be used in a session. It will be applied to all devices created within that session.

*start\_copy* — Indicates whether an SRDF pair should be synchronized after the configuration change is committed.



---

**Note:** When creating SRDF devices, all conversions within a session must have:

- Device configuration settings that reflect the same destination SRDF type (RDF1 or RDF2).
  - The same `ra_group` number.
  - The same `invalidate` option.
  - The same `start_copy` option.
- 

`mvs_ssid` — Specifies the z/OS (MVS) subsystem ID that will be assigned to any device created as a result of removing any mirror(s). If not provided, the original MVS SSID will be assigned when available. If the MVS SSID group is full, you must supply a new MVS SSID. Only one `mvs_ssid` can be used in a session. It will be applied to all devices created within that session.

`raidset` — When requesting to convert a RAID-S group to unprotected devices, set `raidset` equal to `TRUE` and list the first RAID-S member. It is not necessary to list the other members.

## Examples

To convert two existing BCV devices (001C and 001D) to an RDF1-BCV configuration and to invalidate the source R1 and synchronize the SRDF pair, enter in the command file:

```
convert dev 001C:001D to RDF1-BCV, ra_group=1, remote_dev=001c,
invalidate=R1, start_copy=YES;
```

### *RDF devices*

To convert an SRDF device or devices from static SRDF to dynamic RDF, use the following command:

```
convert rdf dev SymDevName[:SymDevName] to dynamic;
```

To convert SRDF device 0014 from static SRDF to dynamic RDF, enter in the command file:

```
convert rdf dev 0014 to dynamic;
```

### *Static thin R1 BCV to dynamic thin R1 BCV*

To convert a static thin R1 BCV device to a dynamic thin R1 BCV, use the following command:

```
symconfigure -sid 397 -nop -v -cmd "convert rdf dev 15EE to dynamic;" commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Session ID      1455360 (0x00163500)
Processing symmetrix 000194900397
{
    convert rdf dev 15EE to dynamic;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Establishing session with Remote cfg srvr (000192600266)..Established.
Locking devices.....Locked.
Initiating COMMIT of Remote configuration changes.....Queued.
Remote COMMIT requesting required resources.....Obtained.
Initiating COMMIT of Local configuration changes.....Queued.
Local COMMIT requesting required resources.....Obtained.
Step 004 of 049 steps.....Executing.
. . .
Remote: Step 032 of 134 steps.....Executing.
Step 025 of 135 steps.....Executing.
```

```
Remote: Step 072, substep 1.....Executing.
Step 071 of 135 steps.....Executing.
Remote: Step 121 of 134 steps.....Executing.
Local: COMMIT.....Done.
Remote: Step 122 of 134 steps.....Executing.
Remote: COMMIT.....Done.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

This requires Solutions Enabler V7.3 and higher.

With Engenuity 5875, a thin non metadvice can be converted to a thin concatenated metadvice, with the system automatically selecting the metas using the `form meta` command. For user-selected meta expansions, use the `add dev` command.

#### *RAID 5 device to BCV*

To convert an existing RAID 5 device to a RAID 5 BCV, use the following form:

```
convert dev DeviceNumber to BCV+R-5;
```

---

**Note:** You can only add the BCV attribute to existing RAID 5 devices; converting non RAID 5 devices to RAID 5 BCVs is not allowed.

---

#### *RAID-S set to unprotected*

To convert a RAID-S set whose members are SRDF devices (either RDF1+R-S or RDF2+R-S) to unprotected SRDF devices, create a command file that requests any member of the RAID-S be converted:

```
convert dev 00D5 to RDF1, raidset=TRUE;
```

Then run the `symconfigure` utility:

```
symconfigure -sid 05605 -file cvt_raids.cmd -v -noprompt commit
```

All members will be converted.

---

**Note:** If there is a problem converting any member of a RAID set, the SYMAPI log file will contain the details.

---

## Restrictions

The following are restrictions for converting devices:

- ◆ The `convert` command can be used for three different classes of device configuration changes, as long as the class types are performed in separate sessions.

---

**Note:** Changes for multiple operation classes can be executed in the same session, except for dynamic SRDF changes and device pool changes.

---

The three class types that cannot be used in the same session are:

- Add/remove BCV/DRV attributes
- Add/remove SRDF attributes
- Increase/decrease mirroring
- ◆ When adding DRV attributes, devices must be unmapped.
- ◆ Full swap operations require the R1 and R2 devices to be the same size.

- ◆ Only the head of a metadevice can have its type changed. The metamember(s) will automatically have the changes applied.
- ◆ You cannot convert one member of a raidset to unprotected without converting all the members to unprotected.
- ◆ When adding/removing SRDF attributes, there are no restrictions on I/O. The SRDF pair must be split or failed over. If failed over, the R1 device must be unmapped.
- ◆ When adding/removing BCV attributes, there are no restrictions on I/O. The standard/BCV pair must be split.
- ◆ The SRDF mode on an SRDF device pairing will be Adaptive Copy Disk by default, unless the SYMAPI\_DEFAULT\_RDF\_MODE is set in the option file. If the device being converted is a diskless R1 device, the RDF mode will default to Adaptive Copy Write Pending, regardless of the option file setting. (Applies to Symmetrix arrays running Enginuity 5876 and higher.)

Symmetrix arrays running Solutions Enabler versions lower than V7.4 continue to set the SRDF mode to synchronous by default.

## Converting thin devices

Thin devices can be converted to other thin device configurations. [Table 10](#) lists the valid conversions.

**Table 10** Thin device conversions

Original thin device <sup>a</sup>	Converted to
TDEV	RDF1+TDEV
TDEV	RDF2+TDEV
TDEV	BCV+TDEV
BCV+TDEV <sup>b</sup>	TDEV
RDF1+TDEV	TDEV
RDF2+TDEV	TDEV
BCV+TDEV	RDF1-BCV+TDEV
BCV+TDEV	RDF2-BCV+TDEV
RDF1+TDEV	RDF1-BCV+TDEV
RDF2+TDEV	RDF2-BCV+TDEV
R1-BCV+TDEV <sup>c</sup>	R1+TDEV
R2-BCV+TDEV <sup>c</sup>	R2+TDEV
R1+TDEV <sup>c</sup>	R1-BCV+TDEV
R2+TDEV <sup>c</sup>	R2-BCV+TDEV

a. The original device type should be created before converting the device.

b. BCV+TDEV to TDEV is the only supported conversion for a BCV+TDEV device.

c. Requires Enginuity 5875 Q32011 SR.

*Examples* To convert a thin device TDEV to an RDF1 thin device:

```
convert dev 0015 to RDF1+TDEV;
```

To create a thin R1 BCV device by converting a thin BCV device:

```
symconfigure -sid 397 -nop -v -cmd "convert dev 15F2 to RDF1-BCV+TDEV ra_group=10
remote_dev=16A invalidate=R1 start_copy=no;" commit
```

## Impact on I/O

The following are restrictions/conditions to avoid impact on I/O activity:

- ◆ The BCV attribute is not allowed on thin SRDF devices or thin dynamic SRDF devices.
- ◆ The device being converted must not be part of a clone session.

## Device group invalid warning

When converting devices that are currently in a device group, some groups will be declared invalid after particular operations.

One example would be if you are performing TimeFinder operations and have placed in a group, one standard (STD) device and one TimeFinder (BCV) device. When `symconfigure` enables you to change the BCV device to a non BCV device, the conversion leaves the group in an invalid state.

There are two ways to handle this situation:

- ◆ Remove the TimeFinder (BCV) device from the group before you begin the conversion process.
- ◆ Or, after the device has been converted, export (`symdg export`) the device group and remove from the file `symdg.txt` the invalid device and import (`symdg import`) the device group.

## Converting a metadvice

To convert a metadvice's configuration from concatenated to striped, use the following command:

```
convert meta SymDevName config=MetaOption
[, stripe_size=MetaStripeSize>[cyl]],
[, protect_data=[TRUE|FALSE],
[bcv_meta_head=SymDevName];
```

Where:

`stripe_size` — Specifies the stripe size in 512 byte blocks.

`protect_data` — Specifies whether the data needs to be protected. Possible values are `TRUE` or `FALSE`. This option can only be used while converting a metadvice from concatenated to striped.

`bcv_meta_head` — Specifies the Symmetrix device name (or thin device name).

Metadevices can be converted from concatenated metadevices to striped devices using the `convert meta` command. The command can also be used to change the stripe size of a striped metadvice.

When converting from concatenated metadevices to striped devices, use the `protect_data` option to preserve data on the BCV metadvice during conversion. The BCV metadvice must match the original standard meta in size, configuration, and member count.

The `bcv_meta_head` option specifies which BCV metadvice to use.

#### *Example*

To convert a concatenated metadvice 0030 to a striped metadvice and preserve data during the conversion using BCV metadvice 01F, enter:

```
convert meta 0030, config=striped, stripe_size=1920,
protect_data=TRUE, bcv_meta_head=01f
```

---

**Note:** The specified BCV metadvice must be identical to the original metadvice in capacity, stripe count, and stripe size.

---

## Thin metadevices

The following thin metadvice conversions are supported:

- ◆ Convert thin concatenated metadevices to thin striped metadevices, without protecting data.
- ◆ Convert thin concatenated metadevices to thin striped metadevices, while protecting data using a regular BCV device.
- ◆ Convert a thin concatenated metadvice to a thin striped metadvice, while protecting the data on the thin meta using a thin BCV device that is persistently allocated.
- ◆ Convert a thin striped metadvice to a thin concatenated device, without protecting the data.

## Restrictions

The following restrictions apply to converting thin metadevices:

- ◆ All existing rules for converting metadevices using BCV devices for data protection apply to thin BCV devices. (such as size, number of members, and member size).
- ◆ A thin BCV device used for data protection must be bound to a thin pool, and must be fully allocated with persistent allocations.
- ◆ RDF BCVs cannot be used for data protection.
- ◆ If a thin BCV metadvice is used for protecting data, the meta head and all members must be bound to a thin pool.
- ◆ All the data on the BCV device used for protecting the data will be lost.

## Setting device attributes

You can set the device attributes or emulation of a number of devices in a range using the following command:

```
set device SymDevName[:SymDevName]
      [emulation=EmulationType]
      [identity = NO identity]
      [attribute=[NO] device_attr];
```

Where:

*emulation* — Specifies the device emulation type, which can be the following: FBA, CELERRA FBA or VME512 FBA.

*identity* — Restores the devices identity to its original value.

*attribute* — Indicates if a device attribute restricts how a device can be accessed. Possible values include:

- *RDB\_Cksum* (Enginuity 5773 and earlier)
- *VCMdb* (for device masking on Enginuity 5773 and earlier)
- *SCSI3\_persist\_reserv* (persistent group reservation) In Enginuity 5874 and lower, this attribute is disabled by default. In Enginuity 5875 and higher, this attribute is enabled by default. For Solutions Enabler V7.5 with Enginuity 5876 Q42012SR and higher, you cannot set or clear the SCSI-3 persistence attribute for mapped devices; you must unmap the device prior to setting or clearing this attribute.
- *dyn\_rdf* (This option provides the most flexibility in performing dynamic SRDF operations)
- *dyn\_rdf1\_only*
- *dyn\_rdf2\_only*
- *ACLX* (for Auto-Provisioning Groups on Enginuity 5874 and higher)
- *RCVRPNT\_TAG* (for RecoverPoint on Enginuity 5875.198.148 and higher)
- *DIF1* (Enginuity 5876 and higher)
- *AS400\_GK* (Enginuity 5876 and higher)

**Note:** Solutions Enabler V7.3 allows the *dyn\_rdf*, *dyn\_rdf1\_only*, and *dyn\_rdf2\_only* attributes to be specified for thin BCV devices.

### Examples

To convert five devices (0015 to 0019) to Celerra FBA emulation, enter in the command file:

```
set device 0015:0019 emulation=CELERRA_FBA;
```

To reset the device attribute of five devices (0015 to 0019) removing *RDB\_Cksum*, enter the following in the command file:

```
set device 0015:0019 attribute=NO RDB_Cksum;
```

## Restrictions

The following restrictions apply when setting device attributes:

- ◆ A device that is mapped or masked to an FCoE port cannot have the *RCVRPVT\_TAG* attribute.

- ◆ When setting the device emulation type, the devices must be unmapped. No I/O to the devices involved.
- ◆ When setting the attribute type to a mapped device, it is recommended that you minimize the I/O activity to the affected devices.

The following restriction apply to setting the DIF1 device attribute:

- ◆ The DIF1 attribute can only be set on FBA devices.
- ◆ DIF1 attribute can be set on both standard and thin host-accessible devices. You cannot set the DIF1 attribute on any internal devices.
- ◆ A device must be unmapped if resetting the DIF1 attribute.
- ◆ A device with the DIF1 attribute can only be mapped to fiber front-end directors (no iSCSI or FCoE).
- ◆ Metadevices with the DIF1 attribute must have the same state, either all set or all reset, on the metahead and all metamembers.
- ◆ DIF1 attribute can not be set on DATA and SAVE devices.
- ◆ Devices can have either the RDB\_Checksum attribute or the DIF1 attribute, not both. The DIF1 flag cannot be set on a device with an active double checksum.
- ◆ Devices can have either ACLX attribute or the DIF1 attribute, not both.
- ◆ There is no relation between the DIF1 attribute and replication. Both source and target devices of any replication can have their own DIF1 setting.
- ◆ The DIF1 attribute needs to be set before requesting a reset. If the reset request is for a device range, and any one of the devices does not have the DIF1 attribute set, an error returns.

The following restrictions apply to setting the AS400\_GK attribute:

- ◆ The AS400\_GK attribute is only supported with Solutions Enabler V7.4 and Enginuity 5876 and higher.
- ◆ The AS400\_GK attribute can only be set on AS/400 or Celerra FBA devices.
- ◆ The device can not be configured as SAVE, DATA, BCV, SRDF or dynamic RDF.
- ◆ The device can be member of an SG, DG, or CG.
- ◆ If an AS/400 or a Celerra FBA device is created using the command `create gatekeeper`, the AS400\_GK attribute is automatically enabled.
- ◆ The AS400\_GK attribute can only be set on non metadevices. This attribute is only supported on AS400 devices with the following emulation types:
  - AS/400\_M2107\_A02
  - AS/400\_M2107\_A04
  - AS/400\_M2107\_A05
  - AS/400\_M2107\_A06
  - AS/400\_M2107\_A07
  - AS/400\_M2107\_A82
  - AS/400\_M2107\_A84
  - AS/400\_M2107\_A85
  - AS/400\_M2107\_A86

- AS/400\_M2107\_A87
- AS/400\_M2107\_050
- AS/400\_M2107\_099
- AS/400\_D910\_099

## Setting device geometry

Solutions Enabler V7.0 and higher provides the ability to set the geometry for a disk on a Symmetrix arrays managed by a Solaris host.

Prior to Symmetrix DMX-4, the block size and sectors per track were reported differently. During a migration from an earlier version, for example from a DMX-3 to a DMX-4, Solaris looks at the new devices and determines that formatting is required (due to the change in block size), and will attempt to wipe the disks by invoking the `format` utility.

By setting the reported device geometry using Solutions Enabler to look like an older Symmetrix array (making a DMX-4 look like a earlier version of DMX, or to emulate a CLARiiON® disk) will satisfy Solaris and allow the migration to proceed. This feature is useful when migrating data from devices residing in earlier Symmetrix arrays to DMX-4 arrays running Enginuity 5773.

---

**Note:** This feature is only for Solaris hosts using the `format` utility.

---

Symmetrix arrays running Enginuity 5773 will display (with `list` and `show` commands) the specified (emulated) device geometry. Therefore, when a device's geometry is changed to emulate an a different device type, the display does not show the native, or true geometry, but the one that was specified.

---

**Note:** Throughout this section the following terms will be used:

`Symm-6` will be used to represent Symmetrix arrays with Enginuity versions 56xx and earlier, and `Symm-7` will represent Symmetrix arrays with Enginuity version 57xx and higher.

---

Some factors to consider when using the new disk geometry feature are:

- ◆ The disk geometry specified will remain as the geometry for the disk, until it is explicitly changed again.
- ◆ If a copy of a target disk is made, you need to remember to change the geometry of the new target too. As an example, if a device is replicated using SRDF from `Symm-6` (source) to `Symm-7`, you may decide to change the geometry on the replicated `Symm-7` device. If the `Symm-7` device were to be cloned, the clone device does not automatically receive the emulated geometry—it must be explicitly set.
- ◆ You need to know the exact geometry of the source disk and need to ensure that before the migration or replication occurs, this geometry is set for the target disk.

---

**Note:** There is also an array-wide setting called `FBA Geometry Emulation` that can only be turned on by an EMC Customer Engineer. When this is set to `Enabled`, all devices on a `Symm-7` will emulate `Symm-6` geometry. When set to `Enabled` on a Symmetrix array



running Enginuity 5874, all devices will emulate Symm-6 geometry. However, even if this setting is enabled, the option of setting a geometry on an individual device level still exists, and this setting will take precedence over the array-wide setting.

This feature is set using the `symconfigure` command, which is executed using a command file. The syntax for setting the device geometry follows:

```
set dev SymDevName[:SymDevName]
  geometry [= SYMM-6 | = SYMM-7 | = CLARIION]
  cyls = n;

set dev SymDevName[:SymDevName]
  geometry = <NO GEOMETRY>;
```

Where:

**SYMM-6** — Refers to the device geometry in Symmetrix arrays running Enginuity 56xx and earlier (64 sectors/track, 15 tracks/cylinder).

**SYMM-7** — Refers to the device geometry in Symmetrix arrays running Enginuity 57xx and higher (128 sectors/track, 15 tracks/cylinder).

**CLARIION** — Refers to the CLARiiON disk geometry (10 sectors/track, 16 tracks/cylinder; each sector is 512 bytes).

***n*** — Refers to the device cylinder size.

**NO GEOMETRY** — Clears the current geometry if the device has an emulated geometry set.

Blocks and sector size for an FBA device is 512 bytes. The device geometry cannot be set on the following devices:

- Metamember
- Vault device
- SFS device
- SAVE device
- DATA device
- DRV
- Mapped device
- Non-FBA device
- VCMDB device

[“Updating a disk label with emulated device geometry” on page 412](#) provides an example of the procedure and system output for updating a disk label for a data migration.

## Showing device geometry

Disk geometry parameters will display if the `-v` option or the `-geometry` option is specified for the following commands:

- ◆ `syndg list -v`
- ◆ `symdev list -v`
- ◆ `sympd list -v`
- ◆ `symld show -geometry`
- ◆ `symdev show -geometry`

◆ `sympd show -geometry`

*Example* The following is an example of the command for displaying the device geometry:

**`symdev -sid 016 show 0317 -geometry`**

```
Device Physical Name      : Not Visible

Device Symmetrix Name     : 0317
Device Serial ID         : N/A
Symmetrix ID             : 000192600175

Number of RAID Groups    : 1
Attached BCV Device      : N/A
Attached VDEV TGT Device : N/A

Vendor ID                : EMC
Product ID               : SYMMETRIX
Product Revision         : 5874
Device WWN               : 60000970000192600175533030333137
Device Emulation Type    : FBA
Device Defined Label Type: N/A
Device Defined Label     : N/A
Device Sub System Id     : 0x1000
Cache Partition Name     : DEFAULT_PARTITION

Device Block Size        : 512

Device Capacity
{
  Cylinders      :      958
  Tracks         :     14370
  512-byte Blocks :    1839360
  MegaBytes      :        898
  KiloBytes      :    919680
}

Effective Device Geometry: Native
{
  Sectors/Track   :        128
  Tracks/Cylinder :        15
  Cylinders       :        958
  512-byte Blocks :    1839360
  MegaBytes       :        898
  KiloBytes       :    919680
}
```

...(output shortened for this example...)

The field Effective Device Geometry can have four possible values:

- ◆ User Defined — Indicates the user has defined a geometry for this device.
- ◆ Native — Indicates the current device geometry is the same as the native geometry for the Symmetrix array.
- ◆ Array wide emulation — Indicates that the array-wide flag for FBA Geometry Emulation is set to Enabled and the effective geometry shown in the output is derived from this setting.

---

**Note:** Even if the array-wide setting is turned on, the user can still define the geometry at the individual device level and this setting will take precedence over the array-wide setting.

---

- ◆ N/A — For non FBA devices.

## Setting device identifiers

Solutions Enabler supports device identifier management on Symmetrix arrays running Engenuity 5773 and higher. This support allows names and identifiers to be defined for Symmetrix devices, HP devices, and VMS devices. The definitions are configured in a command file and processed with the `symconfigure` command.

The format of the command file for setting device identifiers follows:

```
set dev SymDevName[:SymDevName]
[[device_name = 'DevName'] | [NO DevName]]
[[hp_identifier = 'hp_id'] | [NO hp_identifier]]
[[vms_identifier = vms_id] | [NO vms_identifier]];
```

This format accommodates setting a device name or identifier on a single device, a range of devices, or multiple ranges of devices. Device identifiers do not have to be unique for all devices.

### Restrictions for device\_name

- ◆ The `device_name` can be less than or equal to 64 characters in length.
- ◆ Any character may be used except quotes, which are used to mark the start and end of the input.
- ◆ The names are case sensitive.
- ◆ There is no default device name for a device.

### Restrictions for hp\_id

- ◆ The `hp_id` can be less than or equal to 128 characters in length.
- ◆ Any character may be used except quotes which are used to mark the start and end of the input.
- ◆ The identifiers are case sensitive.
- ◆ There is no default HP identifier for a device.

### Restrictions for vms\_id

- ◆ The `vms_id` here is an unsigned 4-byte integer. A `vms_id` can be a number between 0 and 32766. This value will be accepted and displayed as a decimal value.

---

**Note:** The device identifier can include a `device_name` and the `hp_id` or a device name and the `vms_id`. The device identifier cannot include both the `hp_id` and the `vms_id`.

---

## Identifier exclusions

The user cannot set device identifiers on the following devices:

- ◆ Metamembers. The device identifier of the metahead device will display, if applicable.
- ◆ The following internal devices (device identifiers display as N/A):
  - VAULT
  - SFS
  - DRV
  - SAVE (device names will be allowed for this device type)
  - DATA (device names will be allowed for this device type)

## Viewing device identifiers

Device identifiers are viewed through the `symdev list` command. CLI commands are usually limited to an 80-character output, but if the user requests the `hp_id` of a device to be displayed, the line could be over 80 characters in length. A new form of the `symdev list` command displays the device identifiers.

**Note:** This CLI command does not work in offline mode.

The following example commands display each of these identifiers and their sample output:

```
symdev -sid 237 list -identifier device_name -devs 0370:0372
```

```
Symmetrix ID: 000190300237
```

Device			
Sym	Config	Attr	Device Name
0370	2-Way Mir		oracle_database_accounts_device
0371	2-Way Mir		oracle_database_accounts_device
0372	2-Way Mir		oracle_database_accounts_device
0373	2-Way Mir		N/A

```
symdev -sid 237 list -identifier device_nice_name -devs 0370:0373
```

```
Symmetrix ID: 000190300237
```

Device			
Sym	Config	Attr	Nice Name
0370	2-Way Mir		accounts_device
0371	2-Way Mir		accounts_device
0372	2-Way Mir		0001903002370372
0373	2-Way Mir		0001903002370373

```
symdev -sid 237 list -identifier hp_id -devs 0370:0373
```

```
Symmetrix ID: 000190300237
```

Device			
Sym	Config	Attr	HP Identifier
0370	2-Way Mir		hp_oracle_device
0371	2-Way Mir		hp_oracle_device
0372	2-Way Mir		N/A
0373	2-Way Mir		N/A

```
symdev -sid 237 list -identifier vms_id -devs 0370:0373
```

Symmetrix ID: 000190300237

Device			
Sym	Config	Attr	VMS ID
0370	2-Way Mir		8816
0371	2-Way Mir		8817
0372	2-Way Mir		192
0373	2-Way Mir		8819

---

**Note:** Device name, device nice name, HP device identifiers, and VMS device identifiers cannot be used in any control command. All these device identifiers are only for display. Both HP device identifiers and VMS device identifiers can be set on any host and any devices.

---

## Deleting devices

In the command file, you can delete one or more Symmetrix devices from the specified Symmetrix array. Deleting a device frees the space that it once occupied for future use.

To delete one or more devices, use the following command:

```
delete dev SymDevName[:SymDevName][,raidset = TRUE | FALSE];
```

Where:

**raidset** — When requesting to delete a RAID-S group, set **raidset** to **TRUE** and list the first RAID-S member. It is not necessary to list the other members; all will be deleted.

### Examples

To delete Symmetrix device 0015 from Symmetrix array 345, create a command file containing the following:

```
delete dev 0015;
```

Then activate the option using the command:

```
symconfigure -sid 345 -file delete_dev.cmd -v -noprompt commit
```

To delete all devices in a RAID-S set, create a command file with a statement that only identifies the first member of the RAID-S set:

```
delete dev 015C, raidset = TRUE;
```

Then run the **symconfigure** utility:

```
symconfigure -sid 605 -file delete_raids.cmd -v -noprompt commit
```

All members will be deleted.

## Restrictions on deleting devices

The following are restrictions for deleting devices:

- ◆ The device emulation must be one of the following types:

- FBA
- CELERRA\_FBA
- VME\_512\_FBA
- CKD\_3380
- CKD\_3390
- AS400\_*n*

---

**Note:** AS/400 metadevices must be dissolved before being deleted.

---

- ◆ The device must not have an attached BCV or DRV.
- ◆ The device must not have any snap sessions.
- ◆ If the device is a DATA device, it must be disabled and have no used tracks.
- ◆ The device must not be:
  - Mapped to a front-end port
  - Part of an SRDF consistency group
  - A source or target of a clone session
  - Held
  - A virtual device that is in use
  - An RDF, SFS, SAVE, or VCM database device
  - A metamember
  - Masked by VCM
  - Bound if a thin device
  - Part of a storage group or view

## Managing device reservations

With Enginuity version 5670 and higher, you can use the configuration change functionality to reserve devices and front-end mapping addresses for future configuration and masking operations. When using this feature, you reserve the devices/addresses you plan on using, verify that no one else has reserved the resources, and release the reservations when the task is complete.

All reservations are assigned a reserve ID, indicating that the specified devices/addresses are reserved. Any attempt to use the reserved devices/addresses will return a message indicating that the devices/addresses are reserved.

There are two types of device reservations:

- ◆ **Enforced** — Reservations are enforced by the SYMAPI library, and require that you specify the reserve ID to use the devices. This is the default behavior when reserving devices.
- ◆ **Advisory** — Reservations are enforced by co-operating applications. Some applications can ignore advisory reservations, allowing knowledgeable users to make configuration changes on reserved devices, provided that their changes are compatible with the reserving task's goal.

Both types of reservations can have expiration dates associated with them, which will automatically release a reservation if you fail to explicitly do so.

---

**Note:** Device reservations are honored only when devices are explicitly specified during Solutions Enabler configuration change operations. Operations that allow the Enginuity operating environment to choose devices (such as when a meta is formed and only the metahead is specified) do not honor device reservations.

---

Device reservations are enabled (TRUE) by default in the `options` file. To disable device reservations, set the `SYMAPI_ENABLE_DEVICE_RESERVATIONS` parameter in the `options` file to FALSE.

Device reservations are enforced (TRUE) by default in the `options` file. To disable the enforcing of device reservations, set the `SYMAPI_ENFORCE_DEVICE_RESERVATIONS` parameter in the `options` file to FALSE.

---

**Note:** For information on changing the `options` file parameters, refer to the *EMC Solutions Enabler Installation Guide* or the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

---

## Reserving devices

To reserve devices, use the following form:

```
symconfigure -sid SymmID [-expire ExpirationDate]
              [-f[ile] CmdFile | 'redirect stdin' | -cmd "Cmd"]
              -owner Owner -comment UserComment
              [-enforce | -advise]
reserve
```

Where:

*ExpirationDate* — Specifies the date and time for a device reservation to expire. This is an optional parameter, and if not specified, defaults to no expiration. The format for this parameter is:

`[mm/dd/yy][:hh:mm:ss]`

If you only provide the *hh:mm*, the current day will be assumed. If you only provide the *mm/dd*, the current year will be assumed. You can also specify a four-digit year.

*CmdFile* — Specifies the name of any ASCII text file containing a set of commands to process at a higher time. This file can be used in the following ways:

- To reserve devices for specific configuration change operations, in which case the file will list configuration change commands.
- To reserve devices for non-specific operations, in which case the contents of the file will use the following form:

```
reserve dev SymDevName[:SymDevName]
```

Using this method allows you to reserve devices for other applications.

*Owner* — Specifies the name of the owner of the reservation (up to 31 characters long).

*Usercomment* — Indicates a user-specified comment detailing the device reservation (up to 255 characters long).

*-enforce* — Specifies an enforced reservation (default).

*-advise* — Specifies an advisory reservation.

When reserving devices, be sure to note the returned reserve ID for subsequent processing.

*Example* To reserve a set of devices to be deleted at a higher time, enter:

```
symconfigure -sid 3241 -file delete.cmd reserve
-owner "LabMgr" -comment "Deleting RAID-S devices"
-expire 04/15/2005 -enforce
```

where *delete.cmd* contains:

```
delete dev 0015, raidset=true;

delete dev 0017, raidset=true;
```

## Committing changes on reserved devices

When committing changes on devices reserved with the Enforced flag, you must supply the appropriate reserve ID. If you do not have the reserve ID and someone else has reserved the devices, the commit will fail. If you have reserved the devices, or no one else has reserved the devices, then the commit will succeed.

When committing changes on devices reserved with the Advisory flag, some applications may not require a reserve ID. However, the *symconfigure* and *symmask* commands do require a reserve ID.

To commit changes on devices reserved with the *-enforce* option, use the following form:

```
symconfigure -sid SymmID -f[ile] CmdFile commit
[-reserve_id=ResvID[,ResvID[,ResvID]]]
[-remote_reserve_id=ResvID[,ResvID[,ResvID]]]
```

*Example* To commit the changes in the command file *delete.cmd*, enter:

```
symconfigure -sid 3241 -file delete.cmd commit -reserve_id 5
```

## Viewing existing reserved devices

To view existing reservations, use the following commands:

```
symconfigure -reserved list

symconfigure -reserve_id ResvID show
```

Where:

*ResvID* — The device reservation ID.



*Example* To view the set of devices reserved on Symmetrix 54, enter:

```
symconfigure -sid 54 list -reserved
```

The following displays:

Symmetrix ID: 000387940054

S Y M M E T R I X			D E V I C E		R E S E R V A T I O N S	
Reserve ID	Date Reserved	Flags TM	Owner	Devices	Port Addresses	
000001	09/27 11:51	E-	Kevin	01A5	-	
Comment: Testing metadvice routines						
000005	10/10 13:45	EL	JaneH	01B6	01C:0 [C9]	
				01B6	01C:1 [C9]	
Comment: Same device, multiple mapping addresses						
000006	10/11 11:06	E-	SteveM	002E:0030	-	
				0049:004B	-	
				002D	15C:0 [2,0,3]	
				0107	01C:0 [4]	
				014D	01C:1 [25]	
				0004	15C:0 [1,0,4]	
				0047:0048	15C:0 [1,0,5:1,0,6]	
Comment: Set up more devices for RCopy testing						
000008	10/17 15:34	EV	Shiran	01B7:01BA	15C:1 [0,1,0:0,1,3]	
				01BC:01BF	15C:1 [0,1,4:0,1,7]	
Comment: Reserve some devices for VSA port						
000002	10/19 11:27	E-	MikeS	0115:0145	-	
Comment: Deleting RAID-S devices						

Legend for Flags:

Type of Reservation (T) :	E = Enforced	A = Advisory
Mapping Address Mode (M) :	L = Lun addressing	V = Volume Set addressing
	T = Target/lun	C = Channel addressing

To request details on reservation 000008, enter:

```
symconfigure -sid 54 show -reserve_id 8
```

Symmetrix ID: 000387940054

```

Reserve ID           : 000008

Usage                : Mapping
Flags                : Enforced

Reserved Date        : 10/17/2005 15:34:04
Expiration Date       : None

Devices              : 01B7:01BA
Front End Paths
{
  Director Number     : 15C
  Director Port Number : 1
  VSA [VBUS,TID,LUN]  : [0,1,0] - [0,1,3]
}

Devices              : 01BC:01BF
Front End Paths

```

```
{
  Director Number      : 15C
  Director Port Number : 1
  VSA [VBUS,TID,LUN]   : [0,1,4] - [0,1,7]
}
```

```
Owner           : Shiran
Hostname        : api196
Application     : SYMCONFIGURE
User Data       : 0
User Comment    : Reserve some devices for VSA port
```

## Releasing reserved devices

When releasing device reservations, you must supply the appropriate reserve ID. Performing a configuration change on reserved devices will not release them. You must do this as an independent step.

To release reserved devices, use the following form:

```
symconfigure -sid SymmID [-noprompt] -reserve_id
ResvID[, ResvID [, ResvID]]
release
```

Where:

*ResvID*— Specifies the device reservation ID.

*Example* To release the set of devices with the ID 5 and 7, enter:

```
symconfigure -sid 3241 release -reserve_id 5,7
```

# Managing device pools

This section explains how to create and delete device pools, add and remove devices from pools, and enable and disable their members. There are three types of device pools, Snap pools, SRDF/A DSE pools, and thin pools.

TimeFinder/Snap uses SAVE device pools to store pre-update images or change tracks during a virtual copy session. The Symmetrix array supports the creation of multiple named device pools, allowing `symsnap` commands to select a particular pool. This alleviates contention for SAVE device space among several snap sessions and eliminates the possibility of a single session using all pool space.

SRDF uses SRDF/A DSE pools to extend the cache space available to SRDF/A cycles by off-loading some or all of the cycle data from cache to pre-configured disk storage pool. These pools are configured the same as Snap pools.

Thin devices (used for Virtual Provisioning) are bound to DATA device pools. Each write to a thin device by the host is written in the bound DATA device pool. The DATA device pools do not have a default pool.

The following are attributes of device pools:

## All device pools

- ◆ All device pools must have the same disk type (EFD, FC, SATA), unless the `SYMAPI_POOL_ALLOW_MIX_TYPE` options file setting is ENABLE.
- ◆ A device pool can only have devices of one emulation type in them. The pool itself is not defined to be a pool for an emulation type. Instead, the first device added to the pool defines its emulation type.
- ◆ When a pool is created, it will be created as either: `snap`, `rdfa_dse`, or `thin` pool.
- ◆ Device pools are made up of devices that are specifically configured to be used for pools.
- ◆ Devices that do not belong to any pools are available to be added to a pool. Devices in a pool are available for use once they are enabled.
- ◆ Devices can be removed from a pool, but must be disabled and drained.

## Snap pools

A special Snap pool `DEFAULT_POOL` exists by default, which contains SAVE devices that have not been assigned to any named pools but are available for Snap operations.

## SRDF/DSE pools

- ◆ SRDF/A DSE pools can be associated (shared) with multiple SRDF/A groups.
- ◆ A single SRDF/A group can have, at most, four RDFA DSE pools associated with it. There can be at most one FBA, one CKD3390, one CKD3380, and one AS400 pool.
- ◆ An SRDF/A DSE pool can contain only one type of device from the list of FBA, CKD3390, CKD3380, and AS400.

## Thin pools

For information about using Virtual Provisioning and thin pools, refer to [“Creating thin pools” on page 116](#).

---

**Note:** If the Symmetrix array is running Enginuity 5874 and Solutions Enabler V7.1 or higher, a disk group must be specified when creating a SAVE device or DATA device if the `create dev` command is also adding the device(s) to a pool.

---



---

**Note:** For more information about SAVE devices and TimeFinder/Snap operations, refer to the *EMC Solutions Enabler TimeFinder Family CLI Product Guide*.

For information about using SRDF/A DSE pools, refer to the *EMC Solutions Enabler SRDF Family CLI Product Guide*.

---

## Creating a device pool

To create a snap or RDF/A DSE device pool, use the following form:

```
create pool PoolName, type = snap | rdfa_dse
```

The maximum number of device pools per Symmetrix array is 512.

### Example

To create a new pool named `Finance`, enter:

```
create pool Finance, type=snap;
```

## Deleting a device pool

You can only delete a device pool if it does not have any members.

To delete a device pool, use the following form:

```
delete pool PoolName, type= snap | rdfa_dse | thin;
```

### Example

To delete a pool named `Finance`, enter:

```
delete pool Finance, type=snap;
```

## Adding devices to a pool

You can only add disabled and inactive devices to a pool. While adding the devices, you can specify whether they should be enabled or disabled in the pool.

To add devices to a pool, use the following form:

```
add dev SymDevName[:SymDevName] to pool PoolName
type= snap | rdfa_dse | thin
[, member_state=ENABLE | DISABLE];
```

Where:

`member_state` — Indicates whether the device(s) being added should be enabled or disabled in the pool.

### Example

To add and enable devices from `001B` to `001D` to a thin provisioning data pool named `ACT20`, enter:

```
add dev 001B:001D to pool ACT20, type=thin, member_state=ENABLE;
```

## Removing devices from a pool

You can only remove disabled and inactive devices from a pool. Once you remove a device from a pool, the device is available to be added to any other pool.

To remove devices from a pool, use the following form:

```
remove dev SymDevName[:SymDevName] from pool PoolName,
type=snap | rdfa_dse | thin;
```

**Example** To remove device 001E from a pool named HR30, enter:

```
remove dev 001E from pool HR30, type=rdfa_dse;
```

---

**Note:** The last DATA device of a pool in a thin tier cannot be removed, even if that pool has no bound devices.

---

## Enabling/disabling devices

You can enable or disable devices for use in a pool. The devices in a pool do not all have to be in the same state (enabled or disabled). If all the devices in a pool are disabled, then 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 disable a device, all sessions must be terminated, and have no used tracks.

---

**Note:** The last DATA device of a pool in a thin tier cannot be disabled, even if that pool has no bound devices.

---

If a DATA or SAVE device has used tracks and is disabled, the data will be drained to other devices in the pool if there is enough space. When complete, the device state changes to Disabled.

To enable or disable a device in a pool, use the following form:

```
enable | disable dev SymDevName[:SymDevName] in pool PoolName,
type=snap | rdfa_dse | thin;
```

Where:

*PoolName* — Specifies the name of the device pool. It can be from 1 to 12 alphanumeric characters long and include hyphen (-), and underscore (\_) characters.

**Example** To enable the use of three new DATA devices in a pool named Finance, enter:

```
enable dev 01B:01D in pool Finance, type=thin;
```

## Renaming a device pool

To rename a device pool, use the following form:

```
rename pool PoolName to NewPoolName type = <snap | rdfa_dse | thin>;
```

### Restrictions

- ◆ The Symmetrix array must be running Solutions Enabler 7.2 and higher with Enginuity 5875.

- ◆ The new pool name must adhere to the same naming restrictions as when creating a pool.
- ◆ In a single command file that includes operations on a pool and a pool rename, you can:
  - Use the old pool name for the pool operations and rename the pool as the last operation.
  - Rename the pool as the first operation and use the new name for the subsequent pool operations.

You cannot use the old pool name for some operations, change the pool name, and then use the new pool name for subsequent operations.

- ◆ Snap, thin, and DSE pools can be renamed; however, the default pool name (DEFAULT\_POOL) for snap cannot be changed.
- ◆ You cannot create and rename a pool in the same session.

## Monitoring a device pool

To monitor a device pool's usage, the `symcfg monitor` command has been added, as follows:

```
symcfg [-sid SymmID] [-i Interval] [-c Count] [-offline]
      [-percent <1-100> -action ScriptFile [-norepeat]]
      [-snap | -rdfa_dse | -thin] [-pool PoolName]
      [-mb] [-gb]
monitor
```

Where:

`-action` — Selects a script that should be run when the specified percent value is encountered. The full pathname to the action script must be specified. The first argument passed to the script is automatically set to the percent value.

`-c` — Indicates the number (count) of times to display or to acquire an exclusive lock on the Symmetrix host database, the local Symmetrix, and the remote Symmetrix arrays.

`-i` — Specifies the repeat interval in seconds to display or 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.

`monitor` — Checks the total percent full of the devices currently configured in the pool and can optionally execute a script file if a specified percentage is encountered.

`-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.

`-percent` — Causes the action script to be executed when the percent full argument is encountered.

`-pool` — Specifies an SRDF/A DSE, snap, or thin pool.

`-snap` — Specifies that the pool type is a snap pool.

`-rdfa_dse` — Specifies that the pool type is an RDF/A DSE pool.

`-thin` — Specifies that the pool type is a thin pool.

## Viewing a device pool

To display all pools, use the `symcfg list` command with the Symmetrix ID, as shown in the following example command and output:

```
symcfg list -sid 397 -pools
```

```
Symmetrix ID: 000194900397
```

S Y M M E T R I X   P O O L S								
Pool Name	Type	Dev	Dev Config	Total Tracks	Enabled Tracks	Used Tracks	Free Tracks	Full %
DEFAULT_POOL	S M	FBA	2-Way Mir	50775	50775	0	50775	0 Ena
CKD_POOL	T S	3390	2-Way Mir	29376	29376	240	29136	0 Ena
TR1_POOL	T S	FBA	2-Way Mir	3486576	2586576	115260	2471316	4 Ena
testPRC	T -	N/A	Unknown	0	0	0	0	0 Dis
test_prc	T S	FBA	2-Way Mir	14700	0	0	0	0 Dis
HR_POOL	T S	3390	2-Way Mir	146880	146880	960	145920	0 Ena
Total Tracks				3957531	3018099	116820	2901279	3

Legend:

Pool (Type):

S = Snap, R = Rdfa DSE T = Thin

(Tech)nology:

S = SATA, F = Fibre Channel, E = Enterprise Flash Drive, M = Mixed, - = N/A

Some notes on the example output follow:

Note that the pool state is Disabled, if all devices in the pool are disabled. The pool state is enabled, if any device is enabled.

**Free Tracks** is (Enabled - Used) the disabled tracks are not counted when counting free tracks.

**Full %** is (Used Tracks/Enabled Tracks \* 100) disabled tracks are not counted.

The Technology type is new with Enginuity 5875.

## Mapping devices

---

**Note:** With Enginuity 5874, when a device is masked it is automatically mapped. Refer to [Chapter 3, page 3-136](#) for more details.

---

You can map devices to front-end director ports, or a range of devices to consecutive addresses specifying a starting address. This is part of the Symmetrix Device Reallocation (SDR) feature.

To map a Symmetrix device to a director port, use the following command:

```
map dev SymDevName [:SymDevName]to dir DirectorNum:PortNum
[, emulation=CELERRA_FBA]
[starting][target=ScsiTarget,] lun=ScsiLun
[, vbus=FibreVbus]
```

---

Notes: The parameters after the `map dev` command can be entered in any order.

---

The following options are not supported with Enginuity 5874 and higher:

```
[, awwn=awwn | wwn=wwn | iscsi=iscsi]
[, masking host_lun=lun | dynamic_lun];
```

Where:

`awwn` — Specifies the user-given name or alias WWN of a host HBA port, if updating a VCM database. This value is for Enginuity 5773 and earlier.

`dynamic_lun` — Specifies to use the dynamic LUN addressing features but does not require a LUN address for each device. The LUN addresses are assigned based on what may already be in use for that host HBA. This value is for Enginuity 5773 and earlier.

`emulation` — Indicates that the device's emulation type is CELERRA FBA. This option is required when performing operations on a Celerra device, and indicates that you are aware that you are changing the Celerra environment.

`host_lun` — Specifies the LUN addresses to be used for each device that is to be added for the host HBA. This value is for Enginuity 5773 and earlier.

`iscsi` — Specifies the iSCSI name, if updating a VCM database. This value is for Enginuity 5773 and earlier.

`lun` — Specifies the SCSI logical unit number (hex value).

`starting` — Specifies the starting address for the range of devices.

`target` — Specifies the SCSI target ID (hex value).

`vbus` — Specifies the virtual bus (vbus) address for mapping to an FA port if using volume set addressing.

`wwn` — Specifies the unique 64-bit World Wide Name (WWN) identifier for an HBA port, if updating a VCM database. This value is for Enginuity 5773 and earlier.

Mapping a RecoverPoint-tagged device or an AS/400 device to an FCoE director is not supported for Symmetrix arrays running Enginuity 5876 and higher.



*Example: Mapping class*

To map metahead (device) 0030 to director 16A, port 0, and SCSI target/lun 0, 7, enter in the command file:

```
map dev 0030 to dir 16A:0 target=0, lun=7;
```

To obtain a list of used addresses, including the next available address, use the following command:

```
symcfg list -SA all -address -available
```

To map a device 0032 to director 16A, port 0, and SCSI target/lun 0,2 and update the device masking database by specifying the WWN 20000000c920b484 of the host bus adapter (HBA) port through which a host accesses the device, enter the following command:

```
map dev 0032 to dir 16A:0 target=0, lun=2, wwn=20000000c920b484;
```

## Impact on I/O

The following are restrictions/conditions to avoid impact on I/O activity:

- ◆ When mapping, there are no restrictions on I/O if adding a second path.
- ◆ After committing a `symconfigure` mapping operation, you must update the device mapping information within the host system environment. Attempting host activity with a device after it has been removed or altered, but before you have updated the host's device information, can cause host errors. For more information, refer to the *EMC Solutions Enabler SRM CLI Product Guide*.
- ◆ To update your hosts, run the utilities specified for your platform as described in [Appendix E, "Updating the Host."](#) After the host environment is updated, I/O activity can resume with the Symmetrix device.

## Mapping CKD devices

You can map or unmap a range of CKD devices from an EA/EF port. In addition, you can also assign PAV alias addresses for devices already mapped.

### Mapping a range of devices

To map a range of devices to an EA/EF port, use the following form:

```
map dev SymDevName[:SymDevName] to dir
DirectorNum:PortNum, starting
base_address=cuu_address [mvs_ssid=nnn];
```

To map a range of devices to an EA or EF port, using the same addressing from a different port, use the following form:

```
map dev SymDevName[:SymDevName] to dir
DirectorNum:PortNum, copying dir
DirectorNum:PortNum;
```

Where:

**base\_address** — Specifies the base address to be assigned to the first device in the mapping request. It will be incremented by one for each device in the range of devices being mapped.

Mainframe ports expect devices to be mapped in groups to form CU images. The first digit in the address is the CU image number, which can range from 0 to FF (Engenuity 5875). The remaining two digits can range from 00 to 0xFF.

You can divide the set of 256 `cu_address` available in a CU image into base addresses and aliases, as is required.

**mvs\_ssid** — When mapping a range of devices to an EA/EF port, you may need to change the current `mvs_ssid` assigned to the devices. If the devices are becoming part of an existing CU image, they will be assigned the `mvs_ssid` of the devices already mapped. If a new CU image is being formed and mapped, a new `mvs_ssid` can be assigned during the map request.

**copying dir** — Specifies a directory from which to copy; used with Engenuity version 5671 for each EA/EF port.

**Example** To map devices 01D through 023 to CU image A, where the next free base address is 20 and the SSID is 140, enter:

```
map dev 01D:023 to dir 01C:1 starting base_address=A20 mvs_ssid=140;
```

### Copying the mapping of a range of devices

To copy a CU image mapping, all devices in the range must be mapped to the same CU image, or not mapped at all. Devices within the specified range that are not mapped will be ignored as long as they are not mappable (SAVE devices, DRVs, and so on). If a device in the range is mappable, the request will be rejected.

To copy a CU image mapping from one port to another, use the following form:

```
map dev SymDevName[:SymDevName] to dir DirectorNum:PortNum
copying dir DirectorNum:PortNum
```

*Example* To copy addresses of devices 010 through 050 from port 03C:0 to port 03D:0, enter:

```
map dev 010:050 to dir 03D:0 copying dir 03C:0;
```

## Assigning PAV alias addresses to mapped devices

When assigning PAV alias addresses to mapped devices, the aliases will be propagated to all director ports to which the devices are mapped. Devices within the range that are not mapped will be skipped. If any devices in the range are mapped to a different CU image than the first device, an error will be returned. If the device range has base addresses with gaps, the aliases will also have gaps.

To assign PAV alias addresses to devices mapped to EA or EF ports, use one of the following forms, according to the version of Enginuity:

- ◆ For Enginuity version 5671, use the following form:

```
add pav alias to dev SymDevName[:SymDevName], starting
alias=cuu_address;
```

- ◆ For Enginuity version 5771 and higher, use the following form:

```
add pav alias to dev SymDevName[:SymDevName], alias count=nnn;
```

Mainframe ports expect devices to be mapped in groups to form CU images. The first digit in the address is the CU image number, which can range from 0 to 0xF. The remaining two digits can range from 00 to 0xFF.

To assign a PAV alias address range to a CU image, use the following form:

```
add pav alias_range nnn:nnn to mvs_ssid = nnn;
```

*Examples* To add the alias A60 to device 01D, which is already mapped, enter:

```
add pav alias to dev 01D starting alias=A60;
```

To add total PAV aliases to device 37a, enter:

```
add pav alias to dev 37a , alias count=32;
```

To add a PAV alias range to a CU image using SSID 140, enter:

```
add pav alias_range addr 0080:009f to mvs_ssid=140;
```

## Removing PAV alias addresses from mapped devices

To remove PAV alias addresses from devices mapped to EA or EF ports, use the following forms, according to the version of Enginuity:

- ◆ For Enginuity version 5671, use the following form:

```
remove pav alias from dev SymDevName[:SymDevName],
starting alias=cuu_address;
```

- ◆ For Enginuity version 5771 and higher, use the following form:

```
remove pav alias from dev SymDevName[:SymDevName], alias count=nnn;
```

To remove a PAV alias address range from a CU image, use the following form:

```
remove pav alias_range from mvs_ssid =nnn
```

*Example* To remove aliases from CU image 7 so that additional devices can be mapped to it, enter:

```
remove pav alias from dev 012:052, starting alias=750;
remove pav alias from dev 012:052, starting alias=790;
```

To remove PAV alias range from device 37a, enter

```
remove pav alias from dev 37a, alias count=32;
```

To remove PAV alias range from a CU image, enter:

```
remove pav alias_range 0080:009f from mvs_ssid=140;
```

## Unmapping devices

You can unmap devices from front-end directors ports. This is part of the Symmetrix Device Reallocation (SDR) feature.

To unmap devices from a director port, use the following form:

```
unmap dev SymDevName[:SymDevName] from dir
<ALL:ALL | ALL:PortNum |
DirectorNum:ALL | DirectorNum:PortNum>
[, emulation=CELERRA_FBA]
[, devmask_access=remove | retain];
```

Where:

`emulation` — Indicates that the device's emulation type is CELERRA FBA. This option is required when performing operations on a Celerra device, and indicates that you are aware that you are changing the Celerra environment.

`devmask_access` — Indicates whether the device masking database should be updated. The `remove` option indicates that device masking access entries for the device should be removed from the VCMDDB. The `retain` option specifies that entries remain in the database. This option is for Enginuity 5773 and earlier.

## Unmapping a range of devices

Unmapping can be from one or all ports. Since all devices with the same SSID must either be mapped or unmapped, you must provide an SSID when unmapping only some devices in a CU image.

To unmap a range of devices from EA or EF ports, use the following form:

```
unmap dev SymDevName[:SymDevName] from dir
< ALL:ALL | ALL:PortNum | DirectorNum:ALL | DirectorNum:PortNum
[new_ssid=nnn] ;
```

*Examples* To unmap five devices from a CU image so that meta CKDs can replace them, and assign the devices an SSID different from the CU image, enter:

```
unmap dev 13B:13F from dir all:all, new_ssid=0160;
```

To unmap devices 0030 and 0031 from director 16A, port 0, and remove the entries from the device masking database, enter in the command file:

```
unmap dev 0030:0031 from dir 16A:0 devmask_access=remove;
```

## Impact on I/O

The following are restrictions/conditions to avoid impact on I/O activity:

- ◆ When unmapping, no I/O activity is allowed on any devices in the specified mapped path. Devices must be made Not Ready or Write Disabled.

For example, to make the device Not Ready:

```
symdbg create -type [REGULAR | RDF1 | RDF2] DgName
symdbg -g DgName -sid SymmID add dev SymDevName
symdbg -g DgName not_ready
```

If you are unmapping only one path to a multipathed device, you may prefer to write disable that path only:

```
symdbg -g DgName -SA 16A -p 0 write_disable
```

It is possible to make a device Not Ready without creating a device group:

```
symdev -sid SymmID not_ready SymDevName
```

---

**Note:** Do not use the `write_disable` argument with the `symrdf` command, as this write disables the source (R1) device(s) or the target (R2) device(s) to its/their local hosts.

---

- ◆ After committing a `symconfigure` mapping operation, update the device mapping information within the host system environment. Attempting host activity with a device after it has been removed or altered, but before you have updated the host's device information, can cause host errors. For more information, refer to the *EMC Solutions Enabler SRM CLI Product Guide*.

To update your hosts, run the utilities specified for your platform as described in [Appendix E, “Updating the Host.”](#) After the host environment is updated, I/O activity can resume with the Symmetrix device.

## Configuring dynamic RDF-capable devices

---

**Note:** Dynamic SRDF is enabled by default on Symmetrix arrays running Enginuity 5874 and higher. This section applies to Symmetrix arrays running Enginuity 5773 and lower.

---

Devices can be configured with `symconfigure` to be dynamic RDF-capable devices. Dynamic SRDF functionality enables you to create, delete, and swap SRDF pairs while the Symmetrix array is in operation. Using dynamic SRDF technology, you can establish SRDF device pairs from non-SRDF devices, and then synchronize and manage them in the same way as configured SRDF pairs.

In addition, when enabled, you can dynamically swap the SRDF device designations of a specified device group using the `symrdf` command. Source R1 device(s) become target R2 device(s) and target R2 device(s) become source R1 device(s). Swaps using dynamic SRDF perform faster.

---

**Note:** Dynamic swap is not supported if Enginuity versions are mixed between your local and remote Symmetrix arrays.

---

For information on using the devices with SRDF commands, refer to the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*.

The following steps must be performed before dynamic SRDF operations can be performed:

1. The Symmetrix must have the dynamic SRDF feature enabled.
2. The devices on which you want to perform dynamic SRDF operations must have the dynamic SRDF attribute set.

These actions must be performed on both the local and remote Symmetrix arrays.

## Enabling dynamic SRDF on the device

Identify the non SRDF devices that you want available for use as dynamic SRDF devices. Create a command file that enables the dynamic SRDF operations for those devices. The following are possible attributes:

```
dyn_rdf      (Using the dyn_rdf option provides the most flexibility)
dyn_rdf1_only *
dyn_rdf2_only *
```

\* These options limit a device to either an R1 or R2 device, and therefore, using them prevents the ability to perform SRDF swaps.

The following example shows such a file:

```
set dev 0020:002A attribute=dyn_rdf;
set dev 0040:0045 attribute=dyn_rdf;
set dev 0056:005A attribute=dyn_rdf;
```

Then, activate the option using the following command:

```
symconfigure -sid 5605 -file set_device_dyn_rdf.cmd -noprompt -v
commit
```

To view the set of devices currently available for use in dynamic SRDF operations:

```
symdev list -dynamic
```

### Sample output

Symmetrix ID: 000000005605

Device Name		Directors			Device		
		SA	:P	DA :IT	Config	Attribute	Sts (MB)
Cap	Physical						
000A	Not Visible	?????	02B:D0	Unprotected	Grp'd	RW	516
000B	Not Visible	?????	15A:D0	Unprotected	Grp'd	RW	516
000C	Not Visible	?????	01B:D0	Unprotected	Grp'd	RW	516
000D	Not Visible	?????	16A:D0	Unprotected	N/Grp'd	RW	516
000E	Not Visible	?????	02A:D0	Unprotected	N/Grp'd	RW	516
0012	Not Visible	?????	01B:C1	Unprotected	Grp'd	RW	516
0013	Not Visible	?????	16A:C1	Unprotected	Grp'd	RW	516
0014	Not Visible	?????	02A:C1	Unprotected	Grp'd	RW	516
0015	Not Visible	?????	15B:C1	Unprotected	N/Grp'd	RW	516
0016	Not Visible	?????	01A:D1	Unprotected	N/Grp'd	RW	516

For information on using the devices with SRDF commands, refer to the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*.

## Setting SRDF group attributes

RDF group attributes allow you to assign priorities to SRDF/A sessions, and to set the minimum amount of time before attempting an SRDF/A cycle switch.

---

**Note:** RDF attributes for arrays running Solutions Enabler V7.4 can be set using the `symrdf` command. Refer to the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*.

---

To set SRDF group attributes using the `symconfigure` command, use the following form:

```
set [ra | rdf] group GroupNum
    [minimum_cycle_time = CycleTime] |
    [rdf_hw_compression = <ENABLE | DISABLE>] |
    [rdf_sw_compression = <ENABLE | DISABLE>] |
    [rdfa_devpace_autostart = <ENABLE | DISABLE>] |
    [rdfa_dse_autostart = <ENABLE | DISABLE>] |
    [rdfa_dse_pool = <PoolName | NO POOL>
    emulation = [fba | ckd-3390 | ckd-3380 | as400] ] |
    [rdfa_dse_threshold = <% of system WP>] |
    [rdfa_transmit_idle = <ENABLE | DISABLE>] |
    [rdfa_wpace_delay = <delay in usecs>] |
    [rdfa_wpace_threshold = <min % of WP cache>] |
    [rdfa_wpace_autostart = <ENABLE | DISABLE>] |
    [session_priority = SessPriority];
```

Where:

*GroupNum* — The SRDF (RA) group number.

*minimum\_cycle\_time* — The minimum time to wait before attempting an SRDF/A cycle switch. Values range from 1 to 60 seconds for Enginuity V5773 and higher, and 5 to 60 seconds for earlier Enginuity versions. This value can also be set using the `symrdf` command. If this value is set while an RDF consistency group is enabled, you must disable and then re-enable the CG for the setting to take effect.

*rdf\_hw\_compression* — Specifies whether the hardware compression feature is enabled. SRDF hardware compression is only supported on SRDF groups that are defined on fiber RDF directors. Although you can enable/disable SRDF hardware compression on the R2 side, the setting of SRDF hardware compression on the R1 side is what enables or disables the feature.

*rdf\_sw\_compression* — Specifies whether the software compression feature is enabled. This feature can be enabled for Asynchronous and Adaptive Copy mode.

*rdfa\_devpace\_autostart* — Specifies whether the SRDF/A device-level pacing feature is automatically enabled when an SRDF/A session is activated for the SRDF group.

*rdfa\_dse\_autostart* — Specifies whether SRDF/A DSE is automatically activated when SRDF/A session is activated for the group. Valid values are ENABLE or DISABLE. DISABLE is the default.

---

**Note:** You can enable both SRDF hardware compression and SRDF software compression. They work independently.

---

*rdfa\_dse\_pool* — The name of a collection of SAVE devices used for SRDF/A DSE.

*emulation* — The pool emulation type.

`rdfa_dse_threshold` — Specifies the percentage of the Symmetrix array's write pending limit. Once the cache usage of all active groups in the Symmetrix array exceeds this limit, data tracks for this group start to spill over to disks. Valid values are from 20 to 100. The default value is 50.

`rdfa_transmit_idle` — Indicates whether this group has transmit idle support enabled.

`rdfa_wpace_delay` — Specifies the maximum host I/O delay that the SRDF/A write pacing feature will cause. The value is specified in microseconds; the allowable values are from 1 to 1000000 (1 sec). The default value is 50000 usecs (50 ms).

`rdfa_wpace_threshold` — Specifies the minimum percentage of the system write pending cache at which the Symmetrix array will start pacing host write I/Os for this SRDF group. The allowable values are from 1 to 99 percent. The default value is 60%.

`rdfa_wpace_autostart` — Specifies whether the SRDF/A write pacing feature is automatically enabled when an SRDF/A session is activated for the SRDF group.

`session_priority` — The priority used to determine which SRDF/A sessions to drop if cache becomes full. Values range from 1 to 64, with 1 being the highest priority (last to be dropped).

*Examples* To set the SRDF/A session priorities for two different SRDF groups, enter:

```
symconfigure commit -sid 12345 -file setup_rdfa.cmd
```

where `setup_rdfa.cmd` contains:

```
set ra group 24, session_priority=1;
set ra group 42, session_priority=8;
```

To set the RDF/A DSE threshold, associate an SRDF group with a pool, and activate DSE automatically, enter:

```
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=rlpool, emulation=fba;
set rdf group 7 rdfa_dse_autostart=enable;
```

**Note:** Place the command for setting the threshold first in the file.

## Enabling RDF/A

To enable SRDF/A (asynchronous) for a group, use the following form:

```
enable rdfa on ra_group=n,
    make_group_swappable=[TRUE | FALSE];
```

## Disabling RDF/A

To disable SRDF/A (asynchronous) for a group, use the following form:

```
disable rdfa on ra_group=<nnn>,
    delete_support_devices=[TRUE | FALSE];
```



**Example** The following example shows a sample command and the output for listing the SRDF attributes:

```
symcfg list -sid 157 -rdfg 86 -rdfa
```

```
Symmetrix ID : 000192601157
```

```
S Y M M E T R I X   R D F A   G R O U P S
```

```
-----
RA-Grp  Group      Flags    Cycle   Pri Thr Transmit      Write Pacing
      Name      CSRM TDA time      Idle Time (usecs) (%) SAU
-----
 86 (55) RDFDVGR55 -IS- XI.    30  33  50 000:00:00   50000  60 I.-
```

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 :
(S)tatus          : A = Active, I = Inactive, - = N/A
(A)utostart       : X = Enabled, . = Disabled, - = N/A
S(U)pported       : X = Supported, . = Not Supported, - = N/A
```

## Adding/removing SRDF mirrors

Solutions Enabler supports setting both cascaded and concurrent SRDF environments. This feature is set using the `symconfigure` command with `add rdf mirror` and `remove rdf mirror`.

**Note:** All the SRDF features can also be configured using the `symrdf` command.

These commands can set up both dynamic and static SRDF device configurations.

The command file format follows:

```
add rdf mirror to dev SymDevName[:SymDevName]
  ra_group=n, mirror_type = [RDF1 | RDF2],
  remote_dev = SymDevName,
  invalidate = <invalidate_opt>,
  start_copy = [YES | NO]
  [,rdf_mode = [sync|semi|acp_wp|acp_disk|async]];

remove rdf mirror from dev SymDevName[:SymDevName],
  ra_group=n;
```

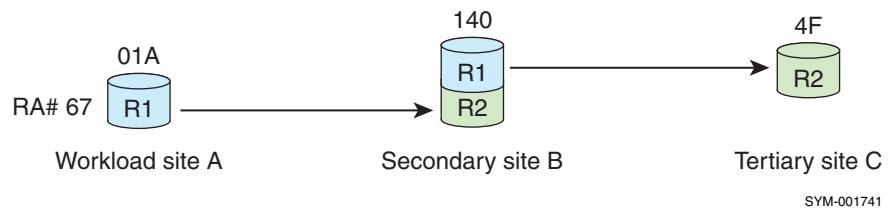
Where:

`invalidate_opt` — Indicates the SRDF device to invalidate so that a full copy can be initiated from the remote mirror. Allowed values are R1 (invalidate the source), or R2 (invalidate the target). The value NONE is allowed if the `start_copy` option is set to NO and the devices are not dynamic SRDF devices.

`rdf_mode` — Sets the SRDF mode when adding a mirror. The default SRDF mode is `acp_disk`, unless the `SYMAPI_DEFAULT_RDF_MODE` is set in the option file. If the R1 device of the created RDF pair is a diskless R1 device, the RDF mode will default to `acp_wp`, regardless of the option file setting. (Applies to Symmetrix arrays running Enginuity 5876 and higher.)

For a client/server environment, the `SYMAPI_DEFAULT_RDF_MODE` option file setting on the server-side host drives the default SRDF mode. The set `rdf_mode` operation on Symmetrix arrays running Solutions Enabler versions earlier than V7.4 continues to set the SRDF mode to `sync` when a mode has not been specified.

*Example* The following example shows how to add a mirror from each site.

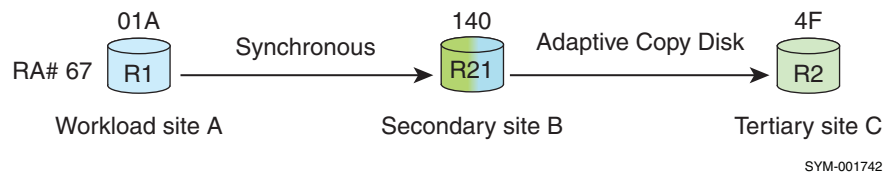


From site A, the `add mirror` command would be as follows:

```
add rdf mirror to dev 01A ra_group=67, mirror_type = RDF1,
remote_dev = 140, invalidate=R1, start_copy = YES,
rdf_mode = sync;
```

From site C, the `add mirror` command would be as follows:

```
add rdf mirror to dev 4F ra_group=67, mirror_type = RDF2,
remote_dev = 140, invalidate=R1, start_copy = YES,
rdf_mode = acp_disk;
```



Possible dual-mirrored device types include the following:

- ◆ R11 — A concurrent R1 device with 2 R1 SRDF mirrors.
- ◆ R21 — An SRDF device with 2 SRDF mirrors. One an R2 mirror, and the other an R1 mirror.
- ◆ R22 — A concurrent R2 device with 2 R2 SRDF mirrors.

With Solutions Enabler V7.3, a mirror can be added to an existing thin BCV to form a thin R1 BCV. BCV devices cannot be concurrent SRDF devices, therefore only one mirror can be added to a thin SRDF BCV device.

## Add/remove mirror restrictions

### R21 devices

The following restrictions apply to adding and removing mirrors for R21 devices:

- ◆ To set up a dynamic R21 device, the device must be both `dyn_rdf1` and `dyn_rdf2` capable. Use the following command to set this value:
 

```
set dev SymDevName[:SymDevName]
    attribute =dyn_rdf;
```
- ◆ When adding the first device to an SRDF group with `-rdf_mode async`, all subsequent devices that are added to the SRDF group must also be added with `-rdf_mode async`.
- ◆ If adding a dynamic SRDF mirror to form an R21 device the second hop (RDF link) can only be in `async` or `acp_disk` mode.
- ◆ If a mode is not specified, the default is synchronous mode.
- ◆ When adding the first device to an SRDF group with `-rdf_mode sync|semi|acp_wp|acp_disk`, subsequent devices cannot be added with `-rdf_mode async`.
- ◆ If the SRDF device is a part of a consistency group, a mirror cannot be added to form an R21 device.
- ◆ Devices cannot be added or removed if there is an active SRDF/A session.
- ◆ RDF mirrors cannot be added to VDEVs, SAVE devices, DATA devices, and thin devices.
- ◆ The command `swap RA group` will not support swapping SRDF groups with devices that have a cascaded or a concurrent SRDF setup.
- ◆ The following conditions must be met when adding a remote mirror to make a device an R21 device:
  - The R21 device should be able to see its current SRDF partner, i.e. the links cannot be partitioned.
  - The remote partner of the R21 cannot be an R21 device or a concurrent SRDF device.

### R22 devices

The following restrictions apply to adding and removing mirrors for R22 devices:

- ◆ R22 devices are only supported on Gig-E and Fibre directors.
- ◆ Creating an R22 device means to add 2 SRDF mirrors. Each SRDF mirror of a single device must belong to a different SRDF group.
- ◆ R22 devices cannot be BCV devices.
- ◆ The two mirrors of the R22 device cannot be paired with the two R1 mirrors of a single concurrent R1 device.
- ◆ RDF mirrors cannot be added to VDEVs, SAVE devices, or DATA devices.
- ◆ An R22 device cannot be converted from static to dynamic.

- ◆ The command `swap ra group` will not support swapping SRDF groups with devices that have a cascaded or a concurrent SRDF setup.

## Static SRDF configuration restrictions

The following restrictions apply to static SRDF configurations:

- ◆ Static SRDF configurations can only be configured in `sync`, `ac_wp`, or `acp_disk` mode.
- ◆ Static SRDF R21 -> R2 leg has to be `acp_disk` mode.
- ◆ The `start_copy` value should be set to `no` and `invalidate` set to `NONE` if adding a remote mirror to a device leads to a concurrent SRDF configuration.

---

**Note:** Currently, the `symconfigure` command enforces a restriction that if `invalidate` is set to `R1` or `R2` `start_copy` should be set to `yes`.

---

For information about control operations for SRDF/Cascaded Replication, refer to the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*.

## Swapping RA groups

You can swap all devices in an RA group from target to source, using the following command:

```
swap ra group nnn,
    refresh=R1|R2, start_copy=YES|NO;
```

Where:

`refresh` — The SRDF device (R1 source or R2 target) to refresh.

`start_copy` — Whether an SRDF pair should be synchronized after the configuration change is committed.

### *Example: Swapping RA group*

To swap RA group 1 from the R1 source to an R2 target group, and then refresh the R2 device, enter:

```
swap ra group 1, refresh=r2, start_copy=yes;
```

## Refresh data concerns

The refresh action identifies 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 copy (refresh) these tracks from the R1 to the R2 device. The result will be the reverse if you use `refresh=R1` as the action.

## Restrictions and I/O impact

RA groups and their device members must be in the correct state before a group swap can be performed. At the group level, SRDF/A and Link Domino cannot be enabled and the group cannot contain a mix of dynamic and static SRDF devices. The following restrictions apply:

- ◆ R1/R2 devices must be the same size.
- ◆ R1 devices cannot have multiple R2 devices.

- ◆ There cannot be any active Open Replicator sessions.
- ◆ Devices must be NotReady, WriteDisabled, or unmapped.
- ◆ When R1 is being refreshed, the R2 cannot have any invalid tracks.
- ◆ When R2 is being refreshed, the R1 cannot have any invalid tracks.
- ◆ Parity RAID devices cannot be swapped.
- ◆ The device cannot be part of an enabled consistency group.
- ◆ Any BCV devices in either group cannot be established.
- ◆ All the devices in the RA group must be of the same type (RDF1 or RDF2) if FarPoint™ is enabled.
- ◆ Only one RA group may be swapped per configuration session.
- ◆ When swapping source and target attributes—no restriction on I/O to R2, but no I/O allowed to the R1 device.
- ◆ When swapping the RA group personalities that engage ESCON directors in a FarPoint connection, be aware that FarPoint buffer settings cannot be adjusted using `symconfigure`. If your FarPoint buffers are set to customized parameters other than default values, an EMC representative will need to be called to adjust the buffer settings after the swap has taken place.

---

**Note:** To swap cascaded or concurrent SRDF groups, use the `symrdf -g DgName swap` command.

---

## Converting directors

With Solutions Enabler V7.4 and Enginuity 5876, you can change the emulation type of a director. This support is for FA and RA director types in a switched SRDF environment. Other director types are not supported. Use the `convert dir` option with the `symconfigure` command to convert an FA director to an RF director, and vice versa, as follows:

```
convert dir DirNum to type = <RF | FA>;
```

---

**Note:** The system checks for Access Control permission CFGSYM and for an SRDF license.

---

### Example

To convert a director type from FA to RF, create a command file containing the following command:

```
convert dir 7H to type = RF;
```

Executing the command file will convert the director type.

```
symconfigure -sid 230 -file convert_dir.cmd commit
```

```
Execute a symconfigure operation for symmetrix '00019490230' (y/[n]) ? y
```

```
A Configuration Change operation is in progress. Please wait...
```

```
Establishing a configuration change session.....Established.
Processing symmetrix 000192600266
```

```

{
    convert dir 16H to type = RF
}
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 007 of 057 steps.....Executing.
. . .
Step 130 of 141 steps.....Executing.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.

The configuration change session has successfully completed.
The configuration change session has completed successfully.

```

After the `convert director` request executes successfully, the following `symcfg list` output can verify the new director types:

```
symcfg -sid 290 list -dir all
```

```
Symmetrix ID: 000194900230
```

S Y M M E T R I X			D I R E C T O R S		
Ident	Symbolic	Numeric	Slot	Type	Status
DF-7A	07A	7	7	DISK	Online
DF-8A	08A	8	8	DISK	Online
DF-7B	07B	23	7	DISK	Online
DF-8B	08B	24	8	DISK	Online
DF-7C	07C	39	7	DISK	Online
DF-8C	08C	40	8	DISK	Online
DF-7D	07D	55	7	DISK	Online
DF-8D	08D	56	8	DISK	Online
FA-7E	07E	71	7	FibreChannel	Online
FA-8E	08E	72	8	FibreChannel	Online
. . .					
<b>RF-7H</b>	<b>07H</b>	<b>119</b>	<b>7</b>	<b>RDF-R1</b>	<b>Online</b>
RE-8H	08H	120	8	RDF-R1	Online

## FA to RF restrictions

The following restrictions apply when converting from FA to RF:

- ◆ The Symmetrix array must not be in SRDF/DM (data mobility) mode.
- ◆ The Symmetrix array must have an SRDF license to convert from FA to RF. However, you can convert from FA to RF without RDF set up on the array.
- ◆ The FA director cannot have mapped devices.
- ◆ The FA / RF conversion must be a standalone request. You cannot execute other commands (such as unmapping devices) and a director FA/RF conversion in the same request.
- ◆ Director types FA and RF are the only supported types for conversion.
- ◆ Solutions Enabler does not enforce director order (allow 7G to be RF if 7H is not).
- ◆ Conversion operations on multiple different directors will be allowed in the same `symconfigure` session, however you cannot convert the same director multiple times in the same session.

- ◆ Solutions Enabler does not support converting the director type of both directors in an SRDF pair in a single session. Separate commands must be issued for converting the directors on local and remote arrays.
- ◆ If an `abort` action is requested when the `commit` is in processing, the abort may fail if the processing has passed the point of no return.
- ◆ Conversion is not allowed when the FA director is a 16 Gb director.

## RF to FA restriction

- ◆ Solutions Enabler will check in the case of RF to FA conversion that the RF director does not have any configured SRDF groups (static or dynamic).

## Setting port characteristics

To set the port characteristics of a specified director, use the following form:

```
set port DirectorNum:PortNum
[FlagName=enable|disable][, ...] ]
gige primary_ip_address=IPAddress
    primary_netmask=IPAddress,
    default_gateway=IPAddress,
    isns_ip_address=IPAddress
    primary_ipv6_address=IPAddress,
    primary_ipv6_prefix=<0 -128>,
[fa_loop_id=Integer] [hostname=HostName];
```

To reconfigure an existing port by copying the port flag settings of another port, use the following form:

```
set port DirectorNum:PortNum,
copying port DirectorNum:PortNum,
[fa_loop_id=Integer] [hostname=<HostName>;
```

When copying the port flag settings from one port to another, both ports must be the same type (Gig-E to Gig-E, FA to FA, and so on). When copying Gig-E ports, IP addresses will not be copied.

---

**Note:** Before you can copy the port flags from one port to another, you must first take the port being copied offline with the `symcfg offline` command.

---

Where:

*FlagName* — A SCSI or fibre port flag. Possible values for the SCSI protocol flags are in [Table 11 on page 104](#), and the values for the fibre protocol flags are in [Table 12 on page 106](#).



**Incorrectly changing the port flags can render your storage system inaccessible. Be sure of your needs before resetting these flags.**

---

*gige* — Indicates that one or more network address values are going to be specified for a front-end Gig-E director. Addresses should use the Internet standard dot notation.

*primary\_ip\_address* — The IP address for a front-end Gig-E port.

**CAUTION**

**When changing the primary IP address, be sure to update all connected hosts with the new address.**

`primary_netmask` — The IP netmask for a front-end Gig-E port.

`default_gateway` — The gateway or router address for a front-end Gig-E port.

`isns_ip_address` — The IP address for the Internet Storage Name Service (ISNS) associated with a front-end Gig-E port.

`primary_ipv6_address` — The IPv6 address for the front-end Gig-E port.

`primary_ipv6_prefix` — The IPv6 mask prefix for a front-end Gig-E port. The value can be 0 - 128, indicating the number of initial bits in the subnet that are identical.

`fa_loop_id` — The FA director loop ID (arbitrated loop physical address). (0 - 125) (Hard Addressing must be enabled.) Not applicable for Gig-E ports.

`hostname` — 12-character hostname.

Table 11 lists the SCSI protocol flags and their descriptions.

**Table 11 SCSI protocol port flags (page 1 of 3)**

SCSI protocol flags	Description
Auto_Busy <sup>a</sup>	When enabled specifically for Unisys A-series platforms only, this flag enables the auto-busy mechanism so that the Symmetrix array returns a Busy to all Unisys host requests.
Avoid_Force_Negotiate <sup>a</sup>	When enabled for Sequent V4.2.3 and lower, the Symmetrix array never initiates negotiations. Normal Symmetrix behavior is to initiate negotiations after an offline-to-online transition. This is for hosts that do not handle negotiations.
Avoid_Reset_Broadcast <sup>a</sup>	When enabled, a SCSI bus reset only occurs to the port that received the reset (not broadcast to all channels).
Command_Reordering <sup>a</sup>	When enabled with Tag Command Queueing in use, the incoming SCSI commands become reordered to Simple Queueing. The default is enabled and should only be disabled upon a request from EMC.
Common_Serial_Number	This flag should be enabled for multipath configurations or hosts that need a unique serial number to determine which paths lead to the same device.
Cyl_Count_In_Name <sup>ab</sup>	When this flag is enabled, the Symmetrix with the specified port embeds the cylinder count into the product ID returned in the SCSI Inquiry command. Enabled for Pyramid only when it is desirable to embed the Symmetrix support into the Pyramid kernel.
Disable_False_Disconnect <sup>ab</sup>	When enabled for debugging, this flag prevents the port from performing a False Disconnect operation. (Default is disabled and currently, you cannot change this flag.)
Disable_Interleaved_Cmds <sup>a</sup>	When enabled (always), metavolume command interleaving is being supported. This allows multiple metamembers to operate at the same time on the same volume.
Disable_Mini_Q <sup>ab</sup>	When enabled for debugging, this flag disables the use of the Mini Queue on the port. (Default is disabled and currently, you cannot change this flag.)
Disable_Q_Reset_on-UA	When enabled, a Unit Attention (UA) that is propagated from another director does not flush the queue for this device on this director. Used for hosts that do not expect the queue to be flushed on a 0629 sense (only on a Hard Reset).
Disable_Ultra <sup>ab</sup>	When enabled, this flag disables Ultra SCSI on an Ultra capable SA port. (Default is disabled and currently, you cannot change this flag.)



**Table 11** SCSI protocol port flags (page 2 of 3)

SCSI protocol flags	Description
Environ_Set	When enabled, this flag enables the environmental error reporting by the Symmetrix to the host on the specific port.
HP3000_Mode <sup>c</sup>	When enabled for HP MPE 5.0 and Enginuity version 5062 and lower, this flag causes the Symmetrix port to return a SCSI Busy state instead of a 0B44 sense code when an xx3C error occurs.
Linked_Commands <sup>ab</sup>	When enabled, this flag enables support of SCSI linked commands. It allows a host to chain SCSI commands in a manner similar to mainframe Channel Command Words (CCWs). (Default is enabled, and currently, you cannot change this flag.)
Negotiate_Reset <sup>d</sup>	When enabled for AS/400 hosts, this flag forces a SCSI negotiation by the Symmetrix array after: <ul style="list-style-type: none"> <li>• A SCSI reset</li> <li>• An error</li> <li>• A bus device reset</li> </ul> This flag is used for AS/400 systems only (default is off).
PBAY_Monitor <sup>a</sup>	For the Sequent platforms only to allow emulation of the Sequent PBAY. When enabled, this flag enables low-level polling of the SCSI bus in order to intercept the nonstandard SCSI operations required for a Sequent PBAY disk subsystem. Must be used for the Sequent cluster operation for the Symmetry system for Sequent V4.2.x operating systems only. Must not be used on versions higher than V4.2.x or for any NUMA-Q systems and also not used for Fibre Channel.
SCSI_3	When enabled, the Inquiry data is altered when returned by any device on the port to report that the Symmetrix supports SCSI 3 protocol. When this flag is disabled, the SCSI 2 protocol is supported.
SCSI_Support	When enabled, this flag provides a stricter compliance with SCSI standards for managing device identifiers, multi-port targets, unit attention reports, and the absence of a device at LUN 0.
Sequent <sup>c</sup>	When enabled, the Symmetrix on the specified port does not force wide or synchronous negotiations, and sets the task timeout window to be 15 seconds before aborting a process. In addition, a busy status is returned instead of an 0B44H when aborting a command on a timeout.
Server_On_AS400	When enabled for AS/400 platforms, this flag indicates that the port is to behave as a server, returning server inquiry data, rather than AS/400 data.
Set_Qerr <sup>a</sup>	This flag should be enabled for SGI platforms only to flush the queue on a contingent allegiance condition (CAC). Must be used for V5.3 and V6.2 SGI operating systems and cluster environments. Not used on versions higher than V6.2.
Siemens <sup>c</sup>	For Siemens R-Series platforms only. When this flag is enabled for Siemens, the Symmetrix array returns in the sense data error 0B48, instead of 0B44 for normal behavior.
Soft_Reset	When enabled for a Bull/GCOS-7 host, the Symmetrix port supports the SCSI Soft Reset option.
SPC2_Protocol_Version <sup>e</sup>	This flag should be enabled (default) in a Windows 2003 environment running Microsoft HCT test version 12.1. When setting this flag, the port must be offline.  <b>Note:</b> Reboot the host after setting this flag.
Sunapee <sup>c</sup>	When enabled for Sun PDB clusters, this flag enables the Sunapee option on the port.

**Table 11** SCSI protocol port flags (page 3 of 3)

SCSI protocol flags	Description
Sync_Transfer <sup>ab</sup>	When enabled, this flag enables SCSI synchronous negotiations. (Default is enabled, and currently, you cannot change this flag.)
Tagged_Commands <sup>ab</sup>	When enabled, this flag enables support for tagged commands. (Default is enabled, and currently, you cannot change this flag.)
Wide_Transfer <sup>ab</sup>	When enabled, this flag enables SCSI Wide operation. (Default is enabled, and currently, you cannot change this flag.)

- a. Supported in Enginuity version 5670 and higher.
- b. Not available for host-based configuration changes.
- c. Obsolete in Enginuity 5874 and higher.
- d. Can only be set on SA ports with Enginuity 5568 and higher.
- e. Requires Enginuity version 5670 and higher.

Table 12 lists the Fibre protocol flags and their descriptions.

**Table 12** Fibre protocol port flags

Fibre protocol flags	Description
ACLX	When enabled, allows storage provisioning using Auto-provisioning Groups. This flag is applicable for Enginuity 5874 and higher.
AS400	This flag should be enabled for any AS400 (iSeries) hosts connecting to the port.
Auto_Negotiate	When enabled, allows two fibre ports to <i>handshake</i> and settle on an optimal speed for data transfer.
Class_2_Service <sup>a,b</sup>	This flag should be enabled for a Class 2 fibre protocol connection that requires an acknowledgement for each frame transmitted. (You <b>cannot change this flag.</b> )
Disk_Array <sup>a</sup>	When enabled (default), the port is represented as a disk array. This port information appears in the Inquiry data.
Generic_VSA <sup>ab</sup>	When enabled, the generic volume set addressing mode is selected. GVSA mode allows hexadecimal addressing. (You <b>cannot change this flag.</b> )
Global__3rdParty_Logout <sup>a</sup>	When enabled (default), an extension is provided to the existing third-party logout required by the standard. In addition to logging out the hosts who are logged in to the port receiving the third-party logout, the logout propagates to other Symmetrix fibre ports that share volumes with the port that had received the logout.
Hard_Address <sup>a</sup>	When enabled (default), the FA director attempts to get the loop_id specified when it initializes on the loop (hard-assigned addressing). When disabled, soft addressing is being used.
Init_Point_to_Point <sup>c</sup>	When enabled, specifies a point-to-point (direct or switched) topology in the initialization sequence. When disabled (default), it is initialized as an arbitrated loop.
Non_Participating <sup>c</sup>	When enabled along with the Hard_Address flag, the Fibre Channel director only uses hard-assigned addressing when it initializes on the loop. Otherwise, soft-assigned addressing is used during loop initialization (the default).
OpenVMS	Enabled for an OpenVMS fibre connection.

**Table 12** Fibre protocol port flags (continued)

Fibre protocol flags	Description
Unique_WWN <sup>c</sup>	When enabled (default) for all environment configuration changes and new environments to ensure unique World Wide Names (WWN) within the fibre environment (uses Symmetrix serial numbers and port numbers). When disabled, you don't have to change WWNs.
VCM_State <sup>d</sup>	Enabled for device masking or the Volume Logix software, which provides volume configuration management controls to handle access to Symmetrix devices. (Disabled is the default.) This flag is only applicable for Enginuity 5773 and earlier.
Volume_Set_Addressing	When enabled along with the Disk_Array flag for HP-UX hosts, the volume set addressing mode is selected. VSA mode allows octal addressing.

- a. Obsolete in Enginuity version 68 and higher.
- b. Not available for host-based configuration changes.
- c. Not available for Gig-E ports.
- d. Obsolete in Enginuity 5874 and higher.

*Example: Setting port characteristics*

To turn on the write protect access logix (ACLX) for director 7E, port 0, enter:

```
set port 7e:0 ACLX=enable;
```

When setting port attributes, it is recommended that you temporarily suspend I/O activity to the effected ports during this operation.

## Enable SYMAPI environment option

Environment option SYMAPI\_CTRL\_OF\_NONVISIBLE\_DEVS in the `options` file must be enabled (or not present in the options file) if there is no device from the local host mapped to this port. For information about enabling this option, refer to [“Enable SYMAPI environment option” on page 30](#).

## Viewing port flags

You can display the possible port flags and their current status with the following command:

```
symcfg -sid 397 list -dir 7e -p 0 -v
```

The following is a partial example of output:

```
Symmetrix ID: 000194900397
Time Zone   : EDT

Product Model          : VMAX-1SE
Symmetrix ID          : 000194900397
...
Director Identification: FA-7E

Director Type          : FibreChannel (563)
Director Status        : Online
...
SCSI Flags
{
  Negotiate_Reset(N)   : Disabled
  Soft_Reset(S)        : Disabled
  Environ_Set(E)       : Disabled
  HP3000_Mode(B)       : Disabled
  Common_Serial_Number(C) : Enabled
  Disable_Q_Reset_on_UA(D) : Disabled
  Sunapee(SCL)         : Disabled
```

```

        Siemens(S) : Disabled
        Sequent(SEQ) : Disabled
        Avoid_Reset_Broadcast(ARB) : Disabled
        Server_On_AS400(A4S) : Disabled
        SCSI_3(SC3) : Disabled
        SPC2_Protocol_Version(SPC2) : Enabled
        SCSI_Support1(OS2007) : Disabled
    }

Fibre Specific Flags
{
    Volume_Set_Addresssing(V) : Disabled
    Non_Participating(NP) : Disabled
    Init_Point_to_Point(PP) : Enabled
    Unique_WWN(UWN) : Enabled
    Access_Logix(ACLX) : Enabled
    OpenVMS(OVMS) : Disabled
    AS400(AS4) : Disabled
    Auto_Negotiate(EAN) : Enabled
...

```

# CHAPTER 2

## Configuring Virtual Provisioning

This chapter describes Virtual Provisioning™ concepts and how to perform operations using the Configuration Change component of the SYMCLI. The chapter covers the following topics:

◆ Virtual Provisioning overview .....	110
◆ Virtual Provisioning (VP) compression .....	111
◆ Creating DATA devices .....	112
◆ Creating thin devices (TDEVs) .....	113
◆ Creating thin pools.....	116
◆ Binding thin devices .....	118
◆ Setting pool attributes .....	121
◆ Allocating space on a thin device .....	123
◆ Space reclamation .....	124
◆ Draining a thin pool .....	126
◆ Automated pool rebalancing .....	129
◆ Viewing thin pools .....	131

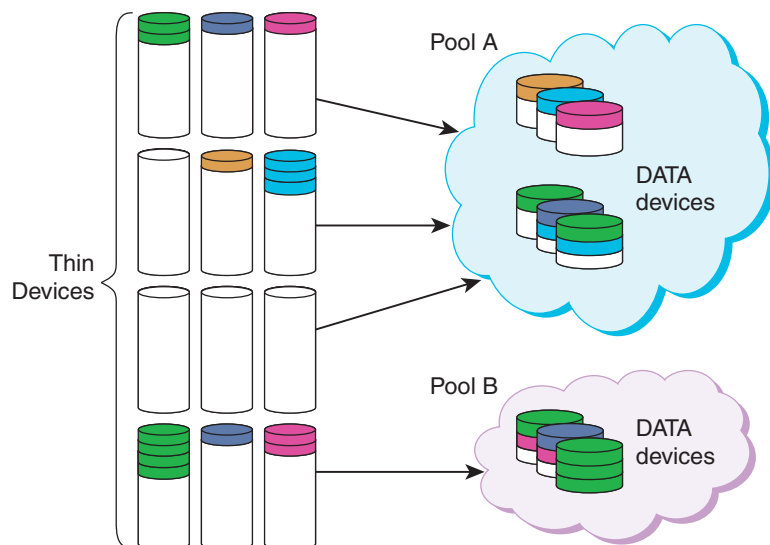
## Virtual Provisioning overview

Virtual Provisioning™ (commonly known as thin provisioning) allows for storage to be allocated and accessed on demand from a pool of storage servicing one or many applications. This type of storage has multiple benefits:

- ◆ Enables LUNs to be “grown” into over time with no impact to the host or application as space is added to the thin pool.
- ◆ Delivers space only from the thin pool when it is written to, that is, on demand. Overallocated application components only use space that is written to—not requested.
- ◆ Provides for thin pool-wide striping and for the most part relieves the storage administrator of the burden of physical device/LUN configuration.

The Virtual Provisioning feature involves these concepts: *thin devices* (TDEV), *DATA devices* (DATADEV), and *thin pools* (THIN). Thin devices can be created with an inflated capacity, because the actual storage space for the data written to the thin devices is on the DATA devices. In this way, when additional storage is needed, more DATA devices can be created in the thin pool.

Figure 6 shows how the data allocated to the thin devices resides in pools of DATA devices. Thin devices bound to pools containing DATA devices are the components used for the Virtual Provisioning feature.



**Figure 6** Data allocation for thin devices

DATA devices are not visible to the host.

Solutions Enabler supports Virtual Provisioning with Open Replicator, SRDF/Synchronous, SRDF/Asynchronous, SRDF Data Mobility (SRDF/DM), TimeFinder/Mirror, TimeFinder/Clone, and TimeFinder/Snap.

The data of a thin device can be spread across multiple pools. The `symcfg` command displays all of the pools in which a thin device has allocated tracks. Refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide* for examples of the `symcfg` displays.

Solutions Enabler V7.4 with Enginuity V5876 supports cascaded storage groups (SGs). Any of the following operations performed on a parent SG will be performed on all devices contained by all child SGs of a specified parent SG:

```
bind, unbind, rebind
start allocate, stop allocate
start free, stop free
start reclaim, stop reclaim
```

Refer to the *EMC Symmetrix Array Management CLI Product Guide* for more information about creating and managing cascaded SGs.

## Virtual Provisioning (VP) compression

Solutions Enabler V7.5 with Enginuity 5876 Q42012 provides a compression feature for thin pools and thin devices. Data can be compressed and uncompressed manually or can be compressed by FAST as part of its processing.

Data that is compressed can be read and written. To allow this, Enginuity uncompresses the data when needed into some space that is reserved in the VP pool that contains the thin devices compressed allocations. To reserve this space and enable allocations in a pool to be compressed, a pool must be enabled for compression. This is done by setting the compression capability on a VP Pool to enabled. When compression is no longer desired for allocations in a pool, the compression capability on a pool can be disabled.

Once a pool is enabled for compression, a background process runs to reserve storage in the pool for temporarily uncompressing data. No compressions of allocations in this pool can occur until this process completes. When compression is disabled, the reserved storage in the pool returns to the pool after all compressed allocations are uncompressed. Disabling the compression capability on a pool does not automatically cause compressed allocations to be uncompressed. This must be done manually. Solutions Enabler reports the current state of the compression capability on a pool, the amount of compressed allocations in the pool, and which thin devices have compressed allocations.

When you compress a device, only allocations that have been written are compressed. Any allocations that were created during the bind or allocate process that have not been written are reclaimed during the compression process. This means that when a compressed device is uncompressed, these allocations will no longer exist. The fact that unwritten allocations can be reclaimed imposes a restriction that persistent allocations, which can not be reclaimed automatically, can not be compressed. In order to allow the compression of persistent allocations, which implies the possible reclaiming of allocations that have not been written, the persistent indicator on the allocations must be manually removed. Note that compression of allocations for a thin device is a background task. Once the compression request is accepted, the thin device will have a background task associated with it that performs the compression. While this task is running, no other background task, like allocate or reclaim, can be run against the thin device.

Compression is set with pool attributes as explained in [“Setting pool attributes” on page 121](#). The commands `symdev compress` and `symdev uncompress` are explained in the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

## Creating DATA devices

DATA devices are similar to SAVE devices, in that they are not visible to the host and must be contained in a pool before they can be used. Thin pools can only contain devices of the same emulation and protection type, however, the DATA devices can be different sizes.

Use a command file (`symconfigure`) to create DATA devices. The syntax related to creating DATA devices is shown below:

```
create dev count=n,
      size = n [MB | GB | CYL],
      emulation=EmulationType,
      config=DeviceConfig
[, attribute=<ckd_meta | savedev | datadev>
  [in pool=PoolName]
  [member_state=<ENABLE | DISABLE> ]
```

Where:

`create dev count` — Specifies the number of DATA devices you are creating.

`size` — Indicates the size of the DATA device. If MB or GB is not specified, the size defaults to cylinders.

`emulation` — Specifies FBA or CKD 3390 devices.

`config` — Specifies the device configuration. The following configuration types are supported for DATA devices: 2-Way-Mir, RAID5 (3+1 and 7+1), and RAID6 (6+2 and 14+2).

`attribute` — Specifies the device type `datadev`.

`in pool` — Specifies the name of the thin pool.

`member_state` — Indicates the state of the DATA device; enable or disable. If a device is disabled, it cannot receive writes.

### DATA device configuration rules

The following configuration rules apply to DATA devices:

- ◆ The only supported emulation types are FBA and CKD 3390.
- ◆ There is no default thin pool.
- ◆ A DATA device cannot be replicated (no BCV, RDF, and so on).
- ◆ A DATA device cannot be a metadevice.
- ◆ DATA devices are not visible to the host; they cannot be mapped or masked.
- ◆ RAID 10 protection for CKD 3390 DATA devices is not supported.



## Creating thin devices (TDEVs)

Thin devices are devices that may or may not have storage allocated to them when they are created. To a host operating system, they look like regular devices with their configured capacity. The host treats them as regular devices and writes and reads from these devices like regular devices.

Use a command file (`symconfigure`) to create thin devices. Solutions Enabler V7.3 and higher supports allocating persistent tracks during the creation of a thin device, binding a thin device to a pool, and during the allocation of a thin device. In addition, reclaiming allocated persistent tracks is also supported.

For thin metadevices, refer to [“Configuring thin metadevices” on page 63](#).

The syntax related to creating thin devices is shown below:

```
create dev count=n,
  size = n [MB | GB | CYL],
  emulation=EmulationType,
  config=DeviceConfig
[, remote_config=DeviceConfig,
  [, binding to pool=PoolName
  [preallocate size = <ALL | n [ MB | GB | CYL]>]
  [allocate_type=persistent]]
  [remote_pool=PoolName]];
```

Where:

`create dev count` — Specifies the number of thin devices you are creating.

`size= n [MB |GB |CYL]` — Specifies the size of each thin device. If MB or GB is not specified, the size defaults to cylinders.

`emulation` — Indicates FBA, CKD 3390, or AS/400 D910\_099 devices.

---

**Note:** Thin CKD devices require Engenuity 5876 or higher.

---

`config` — Specifies TDEV (local thin device), RDF1+TDEV (local SRDF thin device), RDF2+TDEV (remote SRDF thin device), or BCV+TDEV.

`remote_config` — Specifies RDF2. The `remote_config` option is only used when creating both local and remote SRDF thin devices.

`binding to pool` — Specifies the name of the local thin pool.

`preallocate size` — Specifies the amount of space preallocated to the devices. The amount of space must be less than or equal to the available space in the pool and less than or equal to the allocated size of the thin device.

---

**Note:** If you are running Solutions Enabler V7.3, or higher, use the `preallocate ALL` option, which is the entire device.

---

`allocate_type` — An optional parameter that can be specified along with the `preallocate_size` option. With this option, the allocations are unaffected by any reclaim operations, as well as clone, snap, or SRDF copy operations. If it is not supplied, `symconfigure` will preallocate non persistent tracks which can be reclaimed without any additional reclaim flags.

---

**Note:** To remove the persistent allocation, unbind the thin device or execute a `symconfigure start reclaim on dev` command with a persistent type qualifier.

---

`remote_pool` — Specifies the name of the remote thin pool.

## Restrictions when preallocating the size

- ◆ If the ALL option is used, the device size must be less than or equal to the available space in the pool.
- ◆ If a specific preallocate size is mentioned, the space must be less than or equal to the available space in the pool and less than or equal to the specified size of the thin device. If MB or GB is not specified, cylinders are used by default.
- ◆ The preallocate size can only be specified when the newly created device is also being bound to a pool.

## Example of create/bind

To create ten 2 GB thin devices, bind them to thin pool `tp_pool`, preallocate 1 GB to each thin device, and mark the allocations as persistent, enter:

```
symconfigure commit -sid 12345 -file add_new_tdevs.cmd
```

Where `add_new_tdevs.cmd` contains:

```
create dev count=10, size=2GB
    emulation=FBA, config=TDEV,
    binding to pool tp_pool, preallocate size=1 GB
    allocate_type=persistent;
```

Thin devices that were not bound to a data pool during device creation, can be bound using the following form:

```
bind tdev <SymDevName[:SymDevName] |
    in DG DgName | in SG SgName>
    to pool PoolName
    <[preallocate size = <ALL | n [ MB | GB | CYL]>]
    [allocate_type=persistent]>;
```

With Solutions Enabler V7.3 and higher you can preallocate ALL, which is the entire device.

The same metadvice commands (`form meta`, `dissolve meta`, `convert meta`) can be used to configure a thin metadvice. [Table 13](#) provides the supported thin metadvice features.

**Table 13** Supported thin metadvice features

Metadvice feature	Thin device support
Form a thin meta	<ul style="list-style-type: none"> <li>Only thin devices can be used to create a thin metadvice. Mixing of thin and non thin metadvice to create a metadvice is not allowed.</li> <li>Only unbound thin devices can be formed into a metadvice.</li> <li>For a striped metadvice, all members should have the same device size.</li> </ul>
Dissolve a thin meta	Only unbound thin metadvice can be dissolved.
Expand a thin meta	A bound striped or concatenated metadvice can be expanded; however, the new meta member must be unbound. As part of the expansion process, the new metamember will be bound to the same thin pool. The <code>protect_data</code> option is not supported for concatenated metadvice.
Convert a thin meta	Concatenated thin metadvice cannot be converted to striped metadvice. See <a href="#">Table 10 on page 67</a> for supported thin device conversion types.

## Thin device configuration rules

The following rules apply to thin devices:

- ◆ The supported emulation types are FBA, CKD 3390, and AS/400 D910\_099.
- ◆ Thin devices can only be bound to one pool.
- ◆ The thin pool to which the thin devices are bound must have free tracks.
- ◆ A thin device inherits its protection type from the pool to which it is bound. Whatever configuration type the DATA devices have, the thin devices have.
- ◆ Many thin devices can use the same pool.
- ◆ PowerPath® consistency groups are not supported for thin devices.
- ◆ The only supported SRDF configuration is when both the R1 and the R2 are thin devices.

Thin devices can be converted to other thin device configurations. [Table 10 on page 67](#) lists the valid conversions.

## Creating thin pools

Thin pools can be created at the same time as creating the DATA devices. The same naming conventions that apply to SAVE pools and RDF/DSE pools apply. However, there is no default thin pool.

To create a thin pool, use the following form:

```
create pool PoolName, type = thin
[, max_subs_percent= <n>]
[, rebalance_variance = n],
[, max_dev_per_rebalance_scan = n]
[, pool_resv_cap = n]
[, vp_compression = <ENABLE>];
```

Where:

**type** — Specifies the type of device pool. Snap pools are for TimeFinder/Snap operations, SRDF/A DSE pools are for SRDF/A operations, thin pools are for Virtual Provisioning operations with thin devices.

**PoolName** — Specifies the name of the device pool. It can be from 1 to 12 alphanumeric characters long and include hyphens (-), and underscore (\_) characters.

**max\_subs\_percent** — Specifies the maximum total capacity that can be bound to the pool. This can have a value between 0 and 65534. A value of 100 indicates that the total capacity of the thin devices bound to a pool cannot exceed 100 percent of the actual capacity of the pool. A value of 200 indicates that the total capacity of the thin devices bound to a pool cannot exceed twice the amount of the actual capacity of the pool.

A value of 1 will allow only a maximum thin capacity that totals 1/100th the actual physical capacity of the pool. The largest number 65534 indicates that the total capacity of the thin devices bound to a pool can be 65534 percent of the total physical capacity of the pool.

**rebalance\_variance** — Specifies the value (in %) for the pool Rebalance Variance. It can be set in the range of 1 to 50. The default value is 1.

**max\_dev\_per\_rebalance\_scan** — Specifies the maximum limit (in %) for the pool rebalance scan device range. It can be set in the range of 2 to 1024. Its default value is 256.

**pool\_resv\_cap** — (FAST use only) The pool reserved capacity (PRC) is a percentage of the capacity of the virtual pool that will be reserved for non-FAST activities. If the free space in this pool (as a percentage of pool-enabled capacity) falls below the PRC, the FAST controller does not move any more chunks into the pool. Valid PRC values range from 1 to 80 percent.

---

**Note:** PRC values set at the pool-level override any globally set PRC values. For setting a global PRC value for all thin pools, refer to [“Setting FAST control parameters” on page 203](#).

---

`vp_compression` — This option enables compression on the thin pool. This is only a capability setting. Data compresses or uncompresses after a command at the device or group level. Virtual Provisioning compression is explained in [“Virtual Provisioning \(VP\) compression” on page 111](#).

---

**Note:** When creating thin pools it is recommend that you set up thin pool monitoring with the event daemon (`storevntd`) to allow for monitoring of pool out-of-space conditions that could result in application outages.

---

The maximum number of device pools per Symmetrix array is 512.

## Thin pool restrictions

The following restrictions apply to thin pools:

- ◆ The last DATA device cannot be removed from a pool if the pool is enabled for compression.
- ◆ DATA device pools must have the same disk type (EFD, FC, SATA).
- ◆ DATA devices created from external provisioning can only be added to pools of devices created from external provisioning.
- ◆ A thin pool can only contain DATA devices with one emulation type. The pool itself is not defined to be a pool for an emulation type. Instead, the first DATA device added to the thin pool defines its emulation type.
- ◆ All DATA devices in a thin pool must have the same protection. For example, if a thin pool has the first DATA device defined as a 2-Way-Mir, all subsequent added DATA devices must be 2-Way-Mir devices.
- ◆ A DATA device of a specific protection and emulation can be put only into an empty thin pool, or into a pool that *only* contains DATA devices that have the same protection and emulation type.
- ◆ A DATA device can only be removed from a thin pool if it is disabled and has no used tracks on it.
- ◆ DATA devices cannot be disabled and moved from a thin pool in the same session.
- ◆ If the total pool capacity needed to store existing thin device data will be insufficient as a result of a `disable dev SymDevStart:SymDevEnd` command, none of the devices in the requested range will be disabled.

## Binding thin devices

Thin devices are bound to a thin pool (of DATA devices). Thin devices have the following bind operations: bind, unbind, and rebind.

To bind thin devices to a thin pool, use the following form:

```
bind tdev <SymDevName[:SymDevName] |
          in DG DgName | in SG SgName>
to pool PoolName
<[preallocate size = <ALL | n [MB | GB | CYL]>]
[allocate_type = persistent]>;
```

Where:

**preallocate size** — The amount of preallocated space. The space must be less than or equal to the available space in the pool and less than or equal to the specified size of the thin device. If MB or GB are not specified, cylinders are used by default. Solutions Enabler V7.3 introduces the option to preallocate ALL, which is the entire device.

**allocate\_type** — An optional parameter that can be specified along with the **preallocate\_size** option. With this option, the allocations are unaffected by any reclaim operations, as well as clone, snap, or SRDF copy operations. If it is not supplied, **symconfigure** will preallocate non persistent tracks which can be reclaimed without any additional reclaim flags.

---

**Note:** To remove the persistent allocation, unbind the thin device or execute a **symconfigure start reclaim on dev** command with a persistent type qualifier.

---

## Restrictions when preallocating the size

- ◆ If the ALL option is used, the device size must be less than or equal to the available space in the pool.
- ◆ If a specific preallocate size is mentioned, the space must be less than or equal to the available space in the pool and less than or equal to the specified size of the thin device. If MB or GB is not specified, cylinders are used by default.

*Example* The following example binds a thin device to a pool and preallocates the entire device during the bind:

```
symconfigure -sid 266 -nop -v -cmd "bind tdev 0436 to pool
cfg_pool1 preallocate size=all;" commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000192600266
{
  bind tdev 0436 to pool cfg_pool1 preallocate size=150 cyl;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Locking devices.....Locked.
Committing configuration changes.....Started.
Binding devices.....Done.
Allocating devices.....Started.
```

```
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

The following output from a `symdev show` command shows the device capacity:

```
symdev show 0436 -sid 266
```

```
Device Physical Name      : Not Visible
Device Symmetrix Name     : 0436
Device Serial ID         : N/A
Symmetrix ID             : 000192600266

Number of RAID Groups    : 0
Attached BCV Device      : N/A
Attached VDEV TGT Device : N/A

Vendor ID                : EMC
Product ID               : SYMMETRIX
Product Revision         : 5875
Device WWN               : 60000970000192600266533030343336
Device Emulation Type    : FBA
Device Defined Label Type: N/A
Device Defined Label     : N/A
Device Sub System Id     : 0x0004
Cache Partition Name     : DEFAULT_PARTITION
Bound Thin Pool Name     : cfg_pool1

Device Block Size        : 512

Device Capacity
{
  Cylinders          :      150
  Tracks              :      2250
  512-byte Blocks     :     288000
  MegaBytes           :       141
  KiloBytes           :     144000
}
. . . <output abbreviated>
```

## Unbinding thin devices

To unbind thin devices from pools, use the following form:

```
unbind tdev <SymDevName[:SymDevName] |
           in DG DgName | in SG SgName>
           from pool PoolName;
```

A thin device cannot be unbound from a pool if any of the following occurs:

- ◆ Device is mapped to a front-end port or in the Ready state
- ◆ Device is masked by VCM
- ◆ Device has active snap sessions
- ◆ Device is held
- ◆ Device is a source or target of a clone (src or tgt) session
- ◆ Device is a metamember

- ◆ Device is a part of an enabled SRDF CG group
- ◆ Device is an SRDF device
- ◆ Device is under FAST control

## Rebinding thin devices

Previously, if you needed to change the binding of a thin device to a different pool, you had to unbind it and then rebind it to the new pool, which could result in data loss. With Solutions Enabler version 7.2 and higher, `symconfigure` allows you to rebind a thin device to a new pool without moving its data or losing any data. Rebind simply changes the thin device's current binding to a new pool. Rebind will not move any existing data to the new pool, but all new allocations to the thin device will go to the new pool after the rebind action completes.

Use the following command to rebind thin devices to a pool:

```
rebind tdev <SymDevName[:SymDevName] |
           in DG DgName | in SG SgName>
           to pool PoolName;
```

## Restrictions for rebinding thin devices

The following are restrictions for rebinding thin devices:

- ◆ A thin device has to be in the `Bound` state before issuing a rebind.
- ◆ The new binding has to comply with the oversubscription ratio of the new pool. The entire size of the device being rebound will be considered when calculating the oversubscription.
- ◆ If devices in a range, device group, or storage group are bound to different pools, then all the devices will be rebound to the specified pool. No warning/errors will be issued.
- ◆ If a thin device is part of a storage group that is under FAST management, the thin device can only be bound to a pool in a tier that is part of the FAST policy associated with the storage group. Therefore, the device can only be rebound to a pool that is within the policy.
- ◆ If all the devices that are being rebound are already bound to the destination pool, an error returns. If some devices get bound to a pool different than what they are currently bound to, the CLI operation will return a success status.
- ◆ Only one bind, unbind, or rebind operation can be performed on the same device in any one session.

Use the `symdev show` command to check that the pool listed is the pool to which the device is bound, as follows:

```
symdev show -sid 351 1078
```

```
Device Physical Name      : Not Visible
Device Symmetrix Name     : 1078
Device Serial ID         : N/A
Symmetrix ID             : 000194900351
Number of RAID Groups    : 0
```



```

Attached BCV Device      : N/A

Attached VDEV TGT Device : N/A

Vendor ID                : EMC
Product ID               : SYMMETRIX
Product Revision         : 5874
Device WWN               : 60000970000194900351533031303738
Device Emulation Type    : FBA
Device Defined Label Type: N/A
Device Defined Label     : N/A
Device Sub System Id     : 0x0011
Cache Partition Name     : DEFAULT_PARTITION
Bound Thin Pool Name     : gf_thin

Device Block Size       : 512
. . .

```

## Setting pool attributes

You can configure the following attributes for a thin pool:

- ◆ **Maximum subscription percent** — This percent is the maximum limit for the thin pool. This ratio is the total host-perceived capacity of thin devices, divided by physical thin pool capacity.
- ◆ **Pool rebalancing attributes** — Pool Rebalancing is an operation in which uneven usage of data devices on a pool is evened out. With Enginuity version 5875 and Solutions Enabler version 7.2, the `symconfigure` CLI will allow following rebalancing attributes to be set/reset on any thin pool:
  - **Rebalancing variance** — The rebalancing variance is the target device utilization variance for the rebalancing algorithm. The rebalancing algorithm attempts to level distribution of data in a pool so that the percentage utilization of any device in the pool is within the target variance of the percentage utilization any other device in the pool.
  - **Maximum rebalance scan device range** — The maximum rebalance scan device range is the maximum number of devices in a pool on which the rebalancing algorithm will concurrently operate.
- ◆ **Pool reserved capacity** — The pool reserved capacity (PRC) is a percentage of the capacity of the virtual pool that will be reserved for non-FAST activities. If the free space in the pool (as a percentage of pool-enabled capacity) falls below the PRC, the FAST controller does not move any more chunks into the pool.
- ◆ **Compression** — Compression can be enabled when the thin pool is created, or as an attribute on an existing thin pool. When a thin pool is enabled for compression, the data in the pool can be compressed and uncompressed. [“Virtual Provisioning \(VP\) compression” on page 111](#) provides more details.

---

**Note:** VP Compression of CKD devices is not supported on Symmetrix VMAX 10K series arrays.

---

---

**Note:** Rebalancing and compression can only be performed on thin pools and is not applicable for Snap or RDFA/DSE pools.

---

Pool attributes can be set for a thin pool when it is created using the following command form:

```
create pool PoolName,
    type = <snap | rdfa_dse | thin>
    [, max_subs_percent= <n>]
    [, rebalance_variance = <n>]
    [, max_dev_per_rebalance_scan = <n>]
    [, pool_resv_cap = n]
    [, vp_compression = <ENABLE>];
```

Pool attributes can also be set after the thin pool is created using the following form:

```
set pool PoolName,
    type = <thin>,
    <[max_subs_percent= <n | NONE>]
    [, rebalance_variance = <n>]
    [, max_dev_per_rebalance_scan = <n>]>
    [, pool_resv_cap = n | NONE]
    [, vp_compression = <ENABLE | DISABLE>];
```

Refer to [“Managing device pools” on page 83](#) for details about configuring device pools.

**Example** The following example shows the command and output that displays when enabling compression on a VP pool:

```
symconfigure -sid 397 -nop -v -cmd "set pool poolTest, type = thin, vp_compression  
= ENABLE;" commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
{
    set pool poolTest type=thin, vp_compression=ENABLE;
}
```

```
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Committing configuration changes.....Reordering.
Setting pool attributes .....Done.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

## Allocating space on a thin device

Space can be allocated for the thin device through the `preallocate` action during the `bind tdev` command, or using the `start allocate on tdev` command.

The command allows administrators to populate a region of the thin device, or a thin device in a device group or storage group, by giving a starting cylinder on the thin device and the number of cylinders to allocate, as shown:

```
start allocate on tdev <SymDevName[:SymDevName] |
    in DG DgName | in SG SgName>
    start_cyl = n
    <end_cyl = <n | last_cyl> |
    size = <nnn> MB|GB|CYL>
    [allocate_type = persistent];
```

The `start_cyl`, `end_cyl`, and `size` parameters are no longer position dependent. They can be entered in any order.

`allocate_type` — An optional parameter that can be specified along with the `preallocate_size` option. With this option, the allocations are unaffected by any reclaim operations, as well as clone, snap, or SRDF copy operations. If it is not supplied, `symconfigure` will preallocate non persistent tracks which can be reclaimed without any additional reclaim flags.

---

**Note:** If any of the tracks specified for persistent allocation are already allocated, the already allocated tracks will be marked as persistent. The allocation is performed on track groups of 12 tracks, therefore if any track in the track group are persistent, the whole track group becomes persistent, even if some part of the track group was allocated by a command that did not request a persistent allocation.

---

### Examples

The following command starts allocation on the entire device for all the STD thin devices in the DG `myDg`:

```
symconfigure -sid 234 -nop -v -cmd "start allocate on tdev std
    in DG myDg start_cyl=0 end_cyl=last_cyl;" commit
```

The following command starts allocation for all the thin devices in the SG `mySg`:

```
symconfigure -sid 234 -nop -v -cmd "start allocate on tdev in
    SG mySg size=40MB;" commit
```

### Restrictions

- ◆ If a device group is specified, the action of the command is limited to the standard devices in the device group only.
- ◆ Non thin devices in a range, device group, or storage group are ignored (skipped) for any `bind`, `unbind`, `rebind`, `allocate`, and `free` commands.
- ◆ If any of the devices in the device group or storage group are in a state other than bound, the command will fail.
- ◆ If the devices in a device group or storage group are bound to different pools, the `allocate` operation is allowed on the device group or storage group.

The following rules apply to device groups and storage groups with devices of varying sizes:

- ◆ If the allocate options specified are `start_cyl=0` and `end_cyl=last_cyl`, this command will allocate entire device size for all the devices.
- ◆ If `start_cyl` is greater than the size of any device in the device group or storage group, the above command will fail.
- ◆ If the allocate size is specified, and the size is greater than some of the devices, the command will fail.
- ◆ If the allocate options specified is `end_cyl`, and the `end_cyl` specified is larger than the size of any device in the device group or storage group, the above command will fail.

## Space reclamation

With Solutions Enabler V7.3 and higher, two commands, `start reclaim` and `stop reclaim` manage reclaiming thin device tracks.

---

**Note:** The `start free` and `stop free` commands will continue to be supported for unwritten allocations on a range of cylinders for a device, and for stopping the `free` operation on a thin device in the De-allocating state (only).

---

To reclaim tracks on thin devices, use the following form:

```
start reclaim on tdev <SymDevName[:SymDevName] |
in DG DgName | in SG SgName>
[allocate_type = persistent];
```

If the `allocate_type=persistent` is specified, this command frees up tracks that are unwritten or zero-based, even if they are marked as persistent. This is the only way that persistent tracks can be freed using this command.

The command returns a success status if there are no allocations on the specified thin device.

If the allocate type is not specified, this command frees up both unwritten tracks and those that are completely written with zeros. It will not free up tracks that are marked persistent.

## Stop reclamation

To stop the reclaim operation on a thin device, use the following form:

```
stop reclaim on tdev <SymDevName[:SymDevName] |
in DG DgName | in SG SgName>;
```

This command stops a reclaim operation that is currently underway on the thin device.

## Stop free

If you want to free up the unwritten allocated space, use the `start free` command, as follows:

```
start free on tdev <SymDevName[:SymDevName] |
in DG DgName | in SG SgName>
start_cyl = n
<end_cyl = <n | last_cyl> |
```

```
size = nnn MB | GB | CYL>;
```

## Restrictions

- ◆ If a device group is specified, the action of the command is limited to the standard devices in the device group only.
- ◆ If any of the devices in the device group or storage group are not thin devices, then the command will fail.
- ◆ Non thin devices in a range, device group, or storage group are ignored (skipped) for any `bind`, `unbind`, `rebind`, `allocate`, and `free` commands.
- ◆ If the devices in a device group or storage group are bound to different pools, the free operation is allowed on the device group or storage group.
- ◆ Space reclamation cannot be performed while a DATA device in the pool is draining.

## Stopping a background operations on thin devices

Currently, most of the thin device operations run in the background. This means, as an example, that when a user issues a `start reclaim` operation, the command returns as soon as Enginuity receives the request while the actual operation takes place in background. Solutions Enabler displays these background tasks as the thin device status such as `Allocating`, `Deallocating`, `Reclaiming`, and so on.

Previously, Solutions Enabler only allowed users to start these background tasks. With Solutions Enabler 7.1 and higher, users can stop some of these background tasks in the middle of the operation. This feature is only supported for the following actions on thin devices:

- ◆ Stop the free operation on a thin device is the state is `De-allocating`.
- ◆ Stop current allocation process on the thin device.

## Stop free

To stop a free operation, use the following form:

```
stop free on tdev SymDevName[:SymDevName] |
in DG DgName | in SG SgName>;
```

## Restriction

The device has to be in the `De-Allocating` state. An error returns if the device state does not correspond to the requested action.

## Stop allocate

To stop a current allocation process on a thin device, use the following form:

```
stop allocate on tdev <SymDevName[:SymDevName] |
in DG DgName | in SG SgName>;
```

The following command stops allocation on all the STD thin devices in the DG `myDg`:

```
symconfigure -sid 234 -nop -v -cmd "stop allocate on tdev
in DG myDg;" commit
```

The following command stops allocation all the thin devices in the SG `mySg`:

```
symconfigure -sid 234 -nop -v -cmd "stop allocate on tdev  
in SG mySG;" commit
```

## Restrictions

- ◆ A device has to be in the `Allocating` or `Deallocating` state. An error returns if the device state does not correspond to the requested action.
- ◆ If a device group is specified, the action of the command is limited to the standard devices in the device group.
- ◆ Non thin devices in a range, device group, or storage group are ignored (skipped) for any `bind`, `unbind`, `rebind`, `allocate`, and `free` commands.
- ◆ All the devices in the device group or storage group should be in `allocating` or `bound` state. Otherwise, the command will fail.

## Draining a thin pool

**Note:** This feature is for Symmetrix DMX arrays running Enginuity 5773.150 and higher.

Solutions Enabler supports draining and rebalancing thin pools. The `symconfigure` command includes the following commands for this function:

- ◆ `deactivate` — Disables one or more DATA devices without automatic invocation of device draining.
- ◆ `activate` — Enables one or more DATA devices without automatic invocation of stop draining.
- ◆ `start drain` — Starts draining of one or more deactivated DATA devices.
- ◆ `stop drain` — Stops draining of one or more draining DATA devices.

### *Device draining rules*

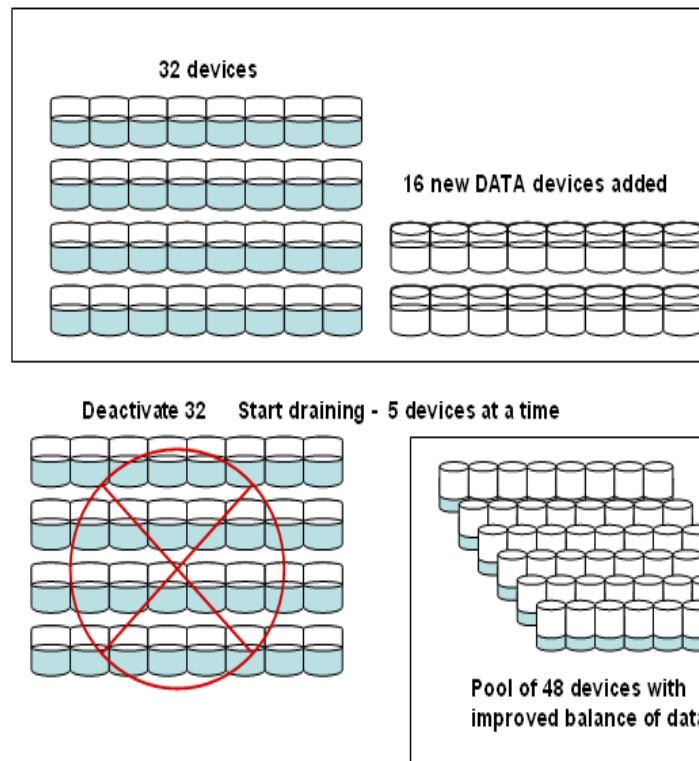
The following rules determine how many devices can be drained:

- ◆ The drain must not cause the enabled devices to end up with greater than 90% utilization in the pool. To calculate this, Solutions Enabler adds the total used tracks on the enabled devices and the total used tracks on the devices that will be drained and divides this sum by the total number of tracks on all the enabled devices. If the result is greater than 90% the drain request is blocked.
- ◆ The number of devices that are draining at any time are limited to 20% total of the number of devices to drain (or draining) plus the number of enabled devices. This limits the impact on the system.

### *Example*

The following is an example of how to use this feature:

As shown in [Figure 7 on page 127](#), you have a pool containing 32 DATA devices; the pool is 50% full. You add 16 new DATA devices to the pool, but you want to spread the pool data so that the additional 16 devices are as populated as the original 32.



**Figure 7** Draining a thin pool

These are the steps you follow to drain the 32 devices to the new 16 and rebalance the data load on the pool:

1. Deactivate 32 of the existing DATA devices. No new allocation will be done on these devices.
2. Perform the following actions until all 32 of the existing devices are processed:
  - a. Start the drain on 5 of the 32 devices—these will be drained to the 16 new DATA devices.
  - b. Monitor until the 5 DATA devices drop a certain percentage in the use.
  - c. Stop the drain.
  - d. Move on to the next set of 5 devices, performing steps a, b, and c for the rest of the original 32 DATA devices.

During this process, any new allocations on thin devices will write to the new 16 DATA devices only, because they are the only ones in an ENABLED state. You will need to factor in these writes while monitoring the drain on the 32 devices.

## Restrictions

The following restrictions apply to this feature:

- ◆ This feature is for thin pools only.
- ◆ All devices in the pools cannot be deactivated at the same time.
- ◆ The drain operation is not supported with any ongoing replication operation.

## Deactivating devices in a thin pool

The following command can be used in a `symconfigure` command file to deactivate devices in a thin pool:

```
deactivate dev SymDevName [:SymDevName]  
in pool PoolName  
type = thin;
```

The `deactivate` command disables the DATA device without starting a drain. This operation puts the DATA device in either of following two states:

**Deactivated** — If the DATA device has any used tracks.

**Disabled** — If the DATA device has zero used tracks.

---

**Note:** The last DATA device of a pool in a thin tier cannot be deactivated, even if that pool has no bound devices.

---

DEACTIVATED state is determined by a combination of the INACTIVE state, the NOT\_DRAINING state, and the presence of used tracks.

Reads and writes can be performed from previously allocated space on deactivated DATA devices. No new allocations can be performed on deactivated DATA devices.

The `deactivate` command is only supported for ENABLED DATA devices.

## Starting the drain on DATA devices

The following command can be used in a `symconfigure` command file to start the drain on the deactivated DATA devices in a thin pool:

```
start drain on dev SymDevName [:SymDevName]  
in pool PoolName  
type = thin;
```

The drain progress can be viewed using the `symcfg` command. It can be viewed by listing the DATA devices in the Symmetrix array or using a detailed pool show.

## Activating devices in a thin pool

The following command can be used in a `symconfigure` command file to activate devices in a thin pool:

```
activate dev SymDevName [:SymDevName]  
in pool PoolName  
type = thin;
```

The `activate` command is essentially same as the `enable` command, however the `activate` command is not allowed if draining is in progress. After the `activate` command, the DATA device will go in the Enabled state.

The `activate` command is only supported for deactivated DATA devices with used tracks.



## Stopping the drain on DATA devices

The following command can be used in a `symconfigure` command file to stop the drain on the DATA devices in a thin pool:

```
stop drain on dev SymDevName [:SymDevName]
in pool PoolName
type = thin;
```

The `stop drain` command can be issued on draining DATA devices to stop the drain operation. After the drain is stopped, the DATA devices are in the Deactivated state.

The `stop drain` command is only supported for draining DATA devices. Devices in any other state return an error.

## Automated pool rebalancing

Whenever writes occur on a thin device, the actual data is written to the DATA devices in the thin pool to which the thin device is bound. The Symmetrix array does spread out the I/O equally on all the DATA devices in the thin pool, therefore, at any given moment the used capacity of each DATA device is uniform across the pool.

However, when new devices are added to the pool, the allocation spread is no longer uniform. With Solutions Enabler 7.1 and higher, you have the ability to automatically rebalance any unbalanced pools.

The `symconfigure` command provides two new actions for starting and stopping a write balancing operation on a per pool basis.

On a `start` operation, Solutions Enabler will:

1. Check to see if a pool balancing operation is in progress for the specified pool.
2. If pool balancing is not in progress, Solutions Enabler will initiate pool balancing on the specified pool.

While write balancing is in progress, it is possible that data will be moved from deactivated DATA devices but never to deactivated DATA devices. The CLI output reports the pool state as `Balancing`. After the balancing operation completes, the pool state will not display `Balancing`. This pool state indicates if any balancing is in progress on a pool at any given time. All pool balancing operations must be initiated by the user.

Stop any ongoing write balancing operation through a CLI command.

Automated pool rebalancing can be invoked on any enabled pool. While pool rebalancing is in progress, all pool operations can still occur. Use the following command within a `symconfigure` command file to start pool rebalancing on a thin pool:

```
start balancing on pool PoolName;
```

### Example

To start pool rebalancing on thin pool `XUN` on Symmetrix 343, enter:

```
symconfigure -sid 343 commit -file balance.cmd
```

where `balance.cmd` contains:

```
start balancing on pool XUN;
```

The `start balancing` operation is only supported on an enabled thin pool. To be enabled, a thin pool must have at least one DATA device in the enabled state.

**Example** To check the status of the thin pool XUN on 343, enter:

```
symcfg show -sid 343 -pool XUN -thin
```

Output similar to the following displays:

```
Symmetrix ID: 000190300343

Symmetrix ID           : 000190300343
Pool Name              : XUN
Pool Type              : Thin
Dev Emulation          : FBA
Dev Configuration      : 2-Way Mir
Pool State             : Balancing
# of Devices in Pool   : 5
# of Enabled Devices in Pool : 5

Enabled Devices(5):
{
-----
Sym      Total      Used      Free  Full  Device  Session
Dev      Tracks     Tracks   Tracks   (%)   State   Status
-----
04B6      8004       2000     6004    25   Enabled  N/A
04B7      8004       2004     6000    25   Enabled  N/A
04B8      8004        500     7504     6   Enabled  N/A
04B9      8004        400     7604     5   Enabled  N/A
04BA      8004       2000     6004    25   Enabled  N/A
-----
Tracks    40020       6459    40020    16
```

## Stopping pool rebalancing

Use the following command within a `symconfigure` command file to terminate a rebalancing operation on a thin pool:

```
stop balancing on pool PoolName;
```

The `stop balancing` operation is only supported on thin pools.

When there is no balancing in progress, the output of the `symcfg pool show` looks similar to the following:

```
Symmetrix ID: 000190300343

Symmetrix ID           : 000190300343
Pool Name              : XUN
Pool Type              : Thin
Dev Emulation          : FBA
Dev Configuration      : 2-Way Mir
Pool State             : Enabled
# of Devices in Pool   : 5
# of Enabled Devices in Pool : 5

Enabled Devices(5):
{
-----
Sym      Total      Used      Free  Full  Device  Session
Dev      Tracks     Tracks   Tracks   (%)   State   Status
-----
04B6      8004       1380     6624    17   Enabled  N/A
```

04B7	8004	1400	6604	17	Enabled	N/A
04B8	8004	1412	6592	18	Enabled	N/A
04B9	8004	1312	6692	17	Enabled	N/A
04BA	8004	1400	6604	17	Enabled	N/A
<hr/>						
Tracks	40020	6459	40020	16		

## Verifying the pool and device states

The `symcfg verify` command allows you to verify the states of DATA devices and thin devices. It also allows you to determine if the pool is in a valid pool state.

To verify if a pool is enabled, disabled, the standard verification options are provided, such as blocking until the pool is in the desired state, and polling at a given rate. To verify a pool state, use the following form:

```
symcfg -sid SymmID [-i Interval] [-c Count]
    <-pool PoolName | -devs <SymDevStart:SymDevEnd |
        SymDevName
        [, <SymDevStart:SymDevEnd | SymDevName>...]>>

verify -datadev
    <-draining | -drainwait | -disabled | -enabled |
    -deactivated | -nonpooled | -balancing>

symcfg -sid SymmID [-i Interval] [-c Count]
    -pool PoolName

verify -poolstate
    <-enabled | -disabled | -balancing>
```

## Viewing thin pools

When you add the `-tdev` option, thin devices display with all pools in which they have allocated tracks. The pool allocated tracks are listed per pool in which the device has allocations. The pool written tracks will only be reported for the bound pool and represents all tracks that have been written to the thin device for any allocation in any pool. The compression is reported by track size (physical size) and the compressed ratio percentage.

The following is an example command and sample output for the `symcfg list -tdev -detail` command:

```
symcfg -sid 397 -range 1620:1630 list -tdev
```

```
Symmetrix ID: 000195700397
```

```
Enabled Capacity (Tracks) : 2884812
Bound Capacity (Tracks) : 20700
```

S Y M M E T R I X    T H I N    D E V I C E S									
<hr/>									
	Bound	Flgs	Total	Total		Total		Compressed	
Sym	Pool Name	EMPT	Tracks	Allocated		Written		Size/Ratio	
				Tracks	(%)	Tracks	(%)	Tracks	(%)
1623	testCust	FX.B	14700	2652	18	744	5	1692	36
1625	-	8...	1500	0	0	0	0	0	0
1626	-	8...	1500	0	0	0	0	0	0
1627	-	8...	1500	0	0	0	0	0	0
162E	testCust	F..C	6000	4800	80	3600	60	1200	75

```

Total
Tracks          -----
                25200      7452  30      4344  17      2892  61

```

## Legend:

```

Flags:  (E)mulation : A = AS400, F = FBA, 8 = CKD3380, 9 = CKD3390
        (M)ultipool : X = multi-pool allocations, . = single pool allocation
        (P)ersistent Allocs : A = All, S = Some, . = None
        S(T)atus    : B = Bound, I = Binding, U = Unbinding, A = Allocating,
                      D = Deallocating, R = Reclaiming, C = Compressing,
                      N = Uncompressing, . = Unbound

```

The emulation flag indicates the device type.

The following example shows the command and output for the `symcfg list -pool -all` command:

```
symcfg list -pool -all -sid 397
```

```
Symmetrix ID: 000195700397
```

```

                S Y M M E T R I X   P O O L S
-----
Pool      Flags  Dev      Usable    Free    Used Full Comp
Name      PTECSL Config  Tracks    Tracks  Tracks  (%)  (%)
-----
DEFAULT_POOL SSFDEI 2-Way Mir    600000    600000      0    0    0
. . .
Pool1      TSFDEI 2-Way Mir    12923748  12945324  105408    0    0
. . .
Mig_src2   TSFEEI RAID-6(6+2)    19224    16824    2400    12    29
Mig_trg2   TSFDEI 2-Way Mir    19224    19188     36    0    0

Total
Tracks          -----
                15638331  15616299  149016    0    2

```

## Legend:

```

(P)ool Type:
  S = Snap, R = Rdfa DSE T = Thin
(T)echnology:
  S = SATA, F = Fibre Channel, E = Enterprise Flash Drive, M = Mixed, - = N/A
Dev (E)mulation:
  F = FBA, A = AS400, 8 = CKD3380, 9 = CKD3390, - = N/A
(C)ompression:
  E = Enabled, D = Disabled, N = Enabling, S = Disabling, - = N/A
(S)tate:
  E = Enabled, D = Disabled, B = Balancing
Disk (L)ocation:
  I = Internal, X = External, M = Mixed, - = N/A

```

The following example shows the output when the pool does contain allocations from devices that are bound to a different pool:

```
symcfg show -pool Mig_trg2 -thin -sid 432 -all -detail
```

```
Symmetrix ID: 000195700432
```

```

Symmetrix ID      : 000194900432
Pool Name         : Mig_trg2
Pool Type         : Thin
Disk Location     : Internal
Technology        : SATA
Dev Emulation     : FBA
Dev Configuration : 2-Way Mir
Pool State        : Enabled

```

```

Compression State           : Enabled
# of Devices in Pool       : 6
# of Enabled Devices in Pool : 6
# of Usable Tracks in Pool  : 19224
# of Allocated Tracks in Pool : 2688
# of Tracks saved by compression : 960
# of Shared Tracks in Pool   : 0
Pool Utilization (%)        : 14
Pool Compression Ratio (%)   : 36
Max. Subscription Percent    : None
Rebalance Variance          : 1%
Max devs per rebalance scan  : 256
Pool Reserved Capacity      : None

```

Enabled Devices(6):

```

{
-----
Sym      Usable      Alloc      Free Full FLG Device
Dev      Tracks      Tracks      Tracks (%) S  State
-----
0811      3204      456      2748  14  .  Enabled
0812      3204      456      2748  14  .  Enabled
0813      3204      444      2760  14  .  Enabled
082A      3204      444      2760  14  .  Enabled
082B      3204      444      2760  14  .  Enabled
082C      3204      444      2760  14  .  Enabled
-----
Tracks    19224      2688      16536  14
}

```

Pool Bound Thin Devices(3):

```

{
-----
Sym  FLG      Pool      Pool      Total      Compressed
Dev  T      Total Sub  Allocated  Written  Size/Ratio
      Tracks (%)  Tracks (%)  Tracks (%)  Tracks (%)
-----
0816  B      4800  25      144  3      0  0      144  0
082F  B      4800  25      1440 30      0  0      720  50
086E  B      750   4      372  50      0  0      252  32
-----
Tracks    10350  54      1956  19      0  0      1116  42
}

```

Other-Pool Bound Thin Devices(2):

```

{
-----
Sym      Bound      Total      Pool      Compressed
      Pool Name  Tracks  Allocated  Size/Ratio
      Tracks      Tracks (%)  Tracks (%)
-----
0826 Mig_src2      6000      360  6      360  0
092C HR_Archive    3000      372  12      252  32
-----
Tracks    9000      732  8      612  16
}

```

Legend:

Enabled devices FLG:

(S)hared Tracks : X = Shared Tracks , . = No Shared Tracks

Bound devices FLG:

S(T)atus : B = Bound, I = Binding, U = Unbinding, A = Allocating,  
D = Deallocating, R = Reclaiming, C = Compressing,  
N = Uncompressing, . = Unbound



# CHAPTER 3

## Masking Devices with Auto-provisioning Groups

This chapter explains how to confine host access to Symmetrix devices using Auto-provisioning Groups and the `symaccess` command of the SYMCLI. The chapter covers the following topics:

◆ Auto-provisioning Groups.....	136
◆ Discovering host HBAs .....	138
◆ Creating groups and views .....	138
◆ Managing masking views .....	144
◆ Managing storage groups .....	149
◆ Managing port groups .....	154
◆ Managing initiator groups .....	156
◆ Displaying Auto-provisioning Group information.....	162
◆ Auto-provisioning Groups examples .....	168

## Auto-provisioning Groups

Auto-provisioning Groups allow storage administrators to create groups of host initiators, front-end ports, and logical devices. These groups are then associated to form a masking view, from which all controls are managed.

---

**Note:** Auto-provisioning Groups are not supported on Symmetrix DMX arrays running Enginuity 5773 and earlier. Storage administrators should continue to use the `symmask` and `symmaskdb` commands to mask devices in Symmetrix DMX arrays.

---

The `symaccess` command provides all the storage provisioning requirements for Symmetrix VMAX Series with Enginuity 5874 and higher.

With `symaccess`, the following device masking functions are no longer supported:

- ◆ There is no device masking database. With the `symaccess` command, all groups and views are backed up to a file, and can be restored from a backup file.
- ◆ Volume visibility and the LUN offset on an initiator are no longer supported.
- ◆ Heterogeneous host types are no longer supported. The HBA flags option is supported.

## How it works

Storage provisioning with `symaccess` allows you to create a group of devices, a group of director ports, a group of host initiators, and with one command, associate them in what is called a *masking view*. Once a masking view exists, devices, ports, and initiators can be easily added or removed from their respective groups.

This feature reduces the number of commands needed for masking devices, and allows for easy management of the masking view.

The `symaccess` command is used to create and manage the groups and views. This command also contains some features similar to those found in the `symmask` and `symmaskdb` commands (a login history command and initiator attributes).

A host-visible (RW) gatekeeper device must be created with the ACLX device attribute. In addition, the ACLX flag must be enabled on the Symmetrix port.

---

**Note:** The `symaccess` command is not supported on Symmetrix DMX arrays.

---

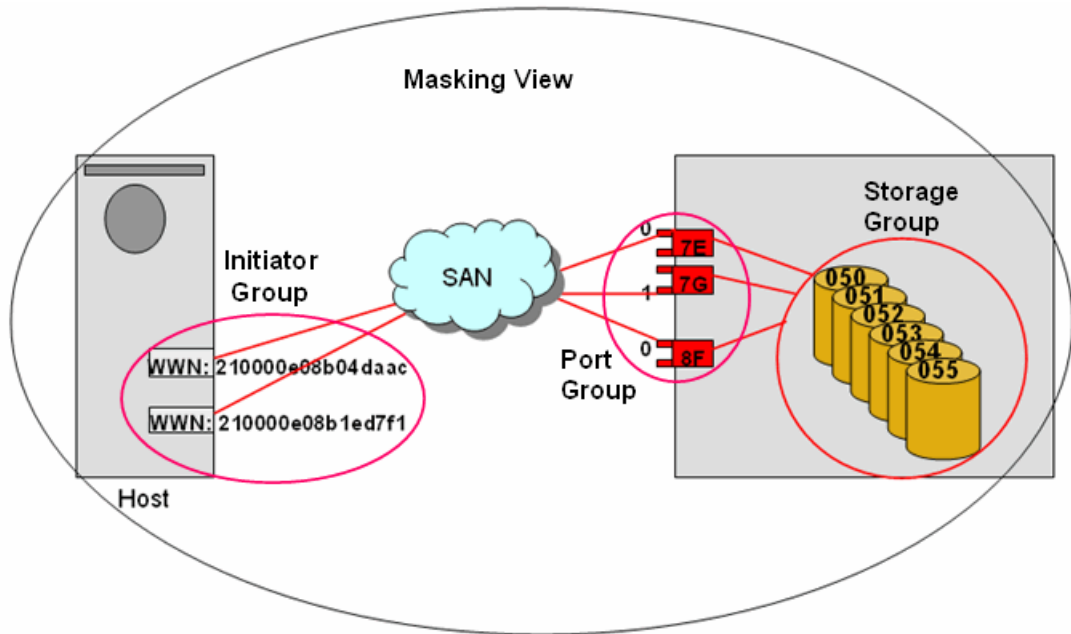
The steps for creating a masking view are:

1. Create a storage group (one or more devices).
2. Create a port group (one or more director/port combinations).
3. Create an initiator group (one or more host WWNs or iSCSIs).
4. Create a masking view containing the storage group, port group, and initiator group.

When a masking view is created, the devices are automatically masked and mapped.

[Figure 8 on page 137](#) shows a masking view.





**Figure 8** Masking view overview

After the masking view is created, any objects (devices, ports, initiators) added to a group automatically become part of the masking view.

## Session rollback

When an Auto-provisioning session fails on Symmetrix arrays running Enginuity 5876 and higher, the system automatically rolls back the ACLX database to the state it was in prior to initiating the session. This rollback feature recovers the database and releases the session lock automatically. The audit log contains any messages relating to the rollback.

**Note:** Storage groups containing CKD devices must already be mapped, and must use the optional flag `-ckd`. Storage groups containing Celerra devices can be masked (and mapped) by using the `-celerra` option. Storage groups containing devices tagged for RecoverPoint can be masked and mapped by using the `-rp` option.

This section covers the following topics:

- [“Discovering host HBAs” on page 138](#)
- [“Creating groups and views” on page 138](#)
- [“Managing masking views” on page 144](#)
- [“Managing storage groups” on page 149](#)
- [“Managing port groups” on page 154](#)
- [“Managing initiator groups” on page 156](#)
- [“Displaying Auto-provisioning Group information” on page 162](#)
- [“Auto-provisioning Groups examples” on page 168](#)

## Discovering host HBAs

During the initial setup, an administrator runs `symaccess discover` on the controlling host to search the environment for Symmetrix devices on each HBA by using the following command:

```
symaccess discover hba
```

---

**Note:** It is assumed that the host from which these commands are run has access to the Symmetrix devices.

---

When the `symaccess discover` command finds a host HBA, it reads the login history table and performs the following:

1. Creates an ASCII alias and writes it to the login history table.

---

**Note:** There is a `-rename` option that can be used with this command to force the discovered hostname/HBA name (or IP address) to be written to the login history table and the initiator group.

---

2. Prints the initiator identifier (WWN/iSCSI) of the HBAs that are connected to the masked channel and Symmetrix array.

The `symaccess discover` command sends information about this connection back to its host system. The `discover` command is the primary mechanism by which hosts other than the control station can learn about their paths to the Symmetrix array. This information can be displayed using the `symaccess list hba` command.

---

**Note:** An iSCSI initiator cannot log in to the array until it belongs to a masking view that includes that specific port on the array.

---

## Creating groups and views

Use the `symaccess` command for:

[“Creating storage groups” on page 138](#)

[“Creating port groups” on page 140](#)

[“Creating initiator groups” on page 141](#)

[“Creating a masking view” on page 142](#)

The management commands for storage groups, port groups, initiator groups, and masking views are explained later in this chapter.

## Creating storage groups

Arrays running Enginuity 5876 and higher can add a storage group to a storage group. This is called *cascaded storage groups*, and is explained in detail in the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*. The `symaccess` command now supports the following operations:

- ◆ Adding child SGs during the creation of an SG

- ◆ Adding a child SG to a parent SG
- ◆ Removing a child SG from a parent SG
- ◆ Backing up and restoring parent SGs
- ◆ Listing cascaded SGs, both parent and child
- ◆ Displaying cascaded SGs, both parent and child
- ◆ Setting Host I/O Limits

You can create a storage group using a range of devices, a list of devices, a device group, a storage group, or a device file. The `symaccess` syntax for creating a storage group is:

```
symaccess -sid SymmID create -name GroupName -type storage
[devs SymDevStart:SymDevEnd |
  SymDevName[, SymDevName[, SymDevName...]] |
  <-g DgName [-std] [-bcv] [-vdev] [-tgt]> |
  <sg SgName[, SgName1, SgName2, . . , SgNameN]>
  <-file DeviceFileName [src] [tgt]>
  [-reserve_id ResvID[, ResvID[, ResvID...]]]
```

You can add devices at the storage group creation time, but those devices cannot have LUNs assigned to them. The LUNs are assigned by the Symmetrix array when the masking view is created.

With Solutions Enabler V7.5, you can set front-end performance controls for a storage group. Front-end performance controls are settings that limit the amount of front-end (FE) bandwidth and I/Os per second (IOPs) that can be consumed by a set of Symmetrix devices over a set of director ports. [“Managing storage groups” on page 193](#) provides additional details, and the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide* provides all the feature details and restrictions.

## Cascaded storage group restrictions

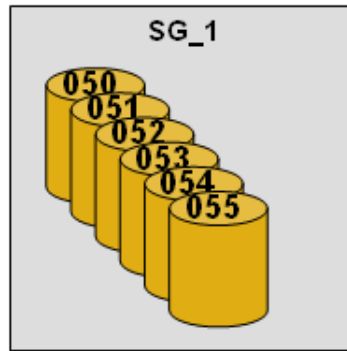
The following restrictions exist for child storage groups (SGs) belonging to a single parent SG:

- ◆ Only a single level of cascading is permitted. A parent SG may not be a child of another SG.
- ◆ SGs can only contain devices or other SGs. No mixing is permitted. This covers attempts to perform device add operations to a parent SG. This also covers attempts to add child SGs to an SG containing devices.
- ◆ A parent can have up to 32 child SGs.
- ◆ A parent SG cannot inherit the same device from more than one child SG.
- ◆ A child SG may only be contained by a single parent SG.

### Example

In the following example, a range of devices containing storage for a critical application are put into a newly created storage group named `SG_1`:

```
symaccess create -sid 458 -name SG_1 -type storage devs 050:055
```



**Figure 9** Storage group SG\_1

Storage group names can be up to 64 characters, and are not case sensitive. Group names must be unique per group *type*, but different group types can share the same name. For example, a storage group, a port group, and an initiator group can all have the name `Financial_DB`. However, two storage groups cannot be named `Financial_DB`.

Device reservations are enforced whenever devices are added to a storage group.

The following sections provide more information about storage groups:

[“Managing storage groups” on page 149](#) explains how to add and remove devices, rename a storage group, and delete a storage group.

[“Displaying Auto-provisioning Group information” on page 162](#) explains the masking view output.

## Creating port groups

Port groups may contain any number of valid front-end ports. A port can belong to more than one port group.

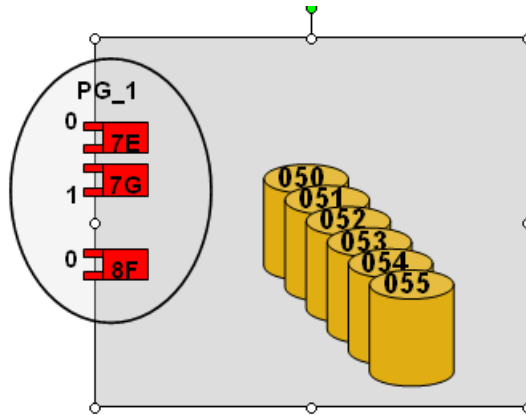
Only Fibre and Gig-E ports on front-end directors will be allowed to be added to a port group. Port groups can have mixed port types.

Ports must have the ACLX flag enabled to be added to a port group. Refer to [“Setting port characteristics” on page 103](#) for information about setting a port flag.

```
symaccess -sid SymmID
  create -name GroupName -type port [-dirport Dir:Port[,Dir:Port...]]
```

**Example** In the following example, a new port group, PG\_1 is created containing three front-end ports:

```
symaccess create -sid 458 -name PG_1 -type port -dirport 7E:0,7G:1,8F:0
```



**Figure 10** Port group PG\_1

The following sections provide more information about port groups:

- ◆ [“Managing port groups” on page 154](#) explains how to add and remove ports, rename a port group, and delete a port group.
- ◆ [“Displaying Auto-provisioning Group information” on page 162](#) explains the masking view output.

## Creating initiator groups

An initiator group is a container of one or more host initiators (Fibre or iSCSI). Each initiator group can contain up to 32 entries. An initiator group may also include the name of another initiator group to allow the groups to be cascaded to a depth of one. An HBA may only belong to one group, but may have masking views for both an upper and lower group if cascaded.

You can create an initiator group using the HBA’s WWN, iSCSI, a file containing WWNs or iSCSI names, or another initiator group name. The `symaccess` syntax for creating an initiator group follows:

```
symaccess -sid SymmID
```

```
create -name GroupName -type initiator [ -wwn wwn | -iscsi iscsi |  
      -file InitiatorFilename | -ig InitiatorGroupName ]  
      [-consistent_lun]
```

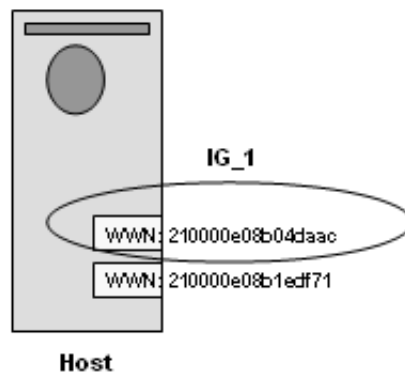
Use the `consistent_lun` option if the devices of a storage group (in a view) need to be seen on the same LUN on all ports of the port group. If the `consistent_lun` option is set on the initiator group, Solutions Enabler will make sure that the host LUN number assigned to devices is the same for the ports on the HBA. If this is not set, then the first available LUN on each individual port will be chosen.

*Example* In the following example, an initiator group, IG\_1 is created containing one WWN:

```
symaccess create -sid 458 -name IG_1 -type initiator -file IG_1
```

Where the file IG\_1 contains:

```
wwn:210000e08b04daac
```



**Figure 11** Initiator group IG\_1

The following sections provide more information about initiator groups:

- ◆ [“Managing initiator groups” on page 156](#) explains how to add and remove initiators, rename an initiator group, and delete an initiator group.
- ◆ [“Displaying Auto-provisioning Group information” on page 162](#) explains the masking view output.

## Creating a masking view

A masking view is a container of a storage group, a port group, and an initiator group. When you create a masking view, the devices in the storage group become visible to the host. The devices are masked and mapped automatically.

Volume dynamic addressing is enabled by default. The Symmetrix array assigns the next available LUN address on the FA port when the masking view is created. The LUN assigned on the FA port will not necessarily match the masking LUN.

The `symaccess` syntax for creating a masking view is as follows:

```
create view -sid SymmID -name ViewName -sg StorageGroupName
           -pg PortGroupName -ig InitiatorGroupName
           [ < [-reserve_id ResvID[, ResvID[, ResvID...]]]
           [-lun Addr] [-ckd] [-celerra] [-rp]
```

The groups being used must already exist and contain some entries (the initiator group can be empty) so that a complete view can actually be created.

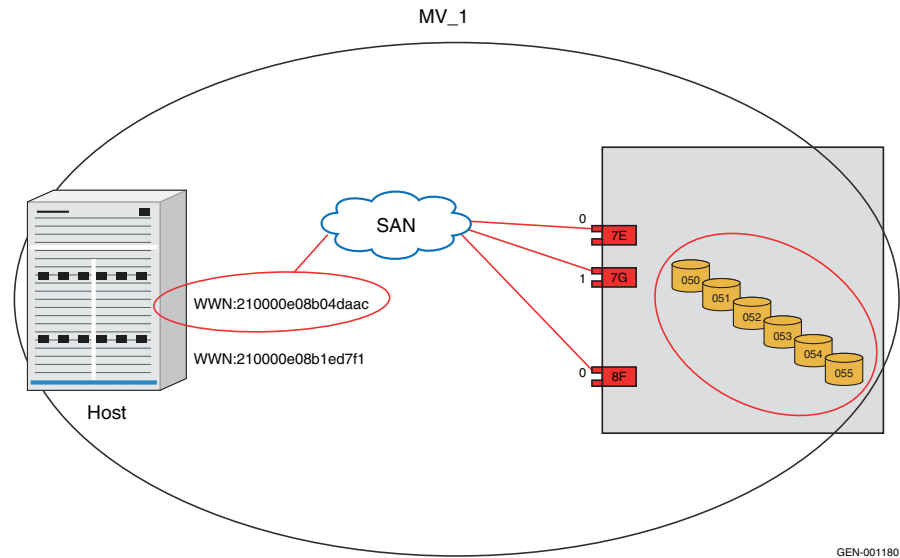
Use the `-ckd` flag to mask CKD devices. By default, CKD devices are blocked from masking but will be allowed if the devices are already mapped. Celerra devices can be masked without the flag if they are not mapped. Use the `-celerra` flag to unmap Celerra devices. Use the `-rp` flag to map or mask devices that have been tagged for RecoverPoint.

When you create a masking view, if Host I/O Limits have been set for the storage group, they become active with the `symaccess create view` command.

### Example

In the following example, a masking view, `MV_1`, is created containing the previously-created storage group, port group, and initiator group:

```
symaccess -sid 458 create view -name MV_1 -sg SG_1 -pg PG_1 -ig IG_1
```



**Figure 12** Masking view MV\_1

After you have created the masking view, if you need to add additional storage for your application, you can add devices to the storage group and they will be automatically masked and mapped in the masking view.

The same principle applies for adding additional front-end ports or host initiators.

If you have storage that you want to manage from the same host, you can create a second storage group and then create a masking view with the same port group and initiator group. The same number of LUNs can be supplied.

Supplying this value is optional and the corresponding input flag should be supplied when it's given.

The masking view shown in [Figure 12 on page 143](#) is a simple example. In a clustered environment, some devices may be seen by the entire cluster, but gatekeeper devices may only need to be seen by individual hosts. To show the flexibility of the storage provisioning functionality, the next section adds additional groups and another masking view to the example.

In this example, two gatekeeper devices are combined into storage group SG\_2:

```
symaccess create -sid 458 -name SG_2 -type storage devs 087,088
```

One of the ports in the port group PG\_1 is placed in a single port group PG\_2:

```
symaccess create -sid 458 -name PG_2 -type port -dirport 7E:0
```

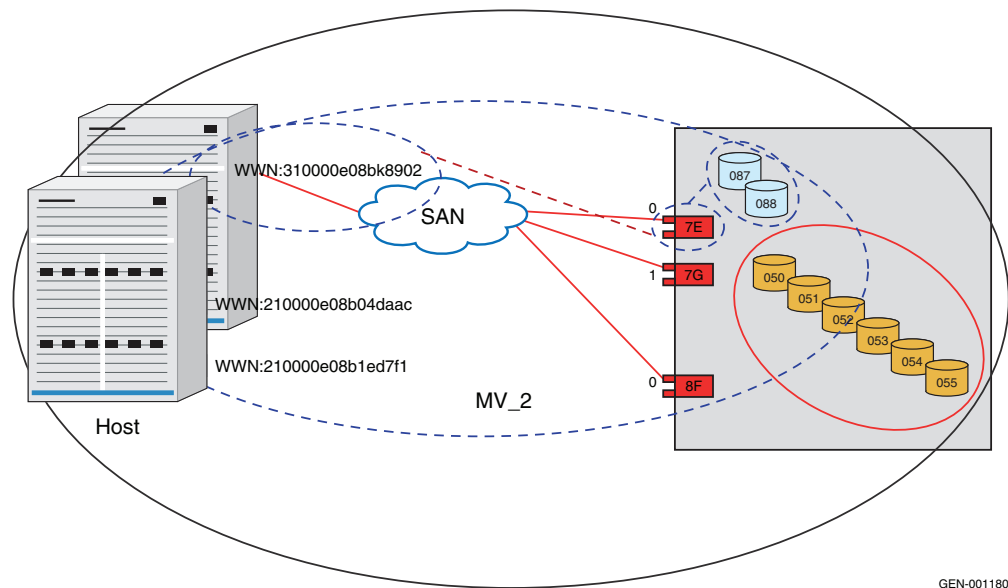
A different host initiator is placed in the initiator group IG\_2:

```
symaccess create -sid 458 -name IG_2 -type initiator -wwn 310000e08bk8902
```

The masking view GKV\_2 is created, with the new groups, as follows:

```
symaccess create view -name GKV_2 -sg SG_2 -pg PG_2 -ig IG_2
```

Figure 13 shows the groups contained in GKV\_2 in the dotted line circles.



**Figure 13** Two masking views

The following sections provide more information about masking views:

[“Managing masking views” on page 144](#) explains how to rename a masking view and delete a masking view.

[“Displaying Auto-provisioning Group information” on page 162](#) explains the masking view output.

## Managing masking views

This section explains how to perform the following actions:

[“Verifying the database” on page 144](#)

[“Deleting masking views” on page 145](#)

[“Naming groups and views” on page 145](#)

[“Backing up and restoring views” on page 146](#)

## Verifying the database

Use the `symaccess verify` command to verify that the Auto-provisioning database is consistent. Any inconsistencies display in the command output. This command can also be used with a backup file. Add the `-log` option for reporting the inconsistencies in a log file and on the screen.

To verify the ACLX database, use the following form:

```
symaccess -sid SymmID | -file BackupFileName
verify [-log]
```



*Example s* The following example shows the message displayed when no consistencies are found:

```
symaccess -sid 266 verify
```

```
Starting a verify operation.....
The auto provisioning database is consistent
```

The next example shows verification of a database backup file with inconsistencies:

```
symaccess -file /tmp/bkup1.file verify
```

```
Starting a verify operation.....
```

```
Found SG 'stor_grp1' to contain the view flag but didn't find a matching view
```

```
Found IG 'init_grp1' contains invalid initiator records
```

```
Found masking view 'mask_view1' with parent IG 'init_grp1' but no masking records
for the child IG 'child_grp1' are present
```

```
There are inconsistencies in the auto provisioning database
```

## Deleting masking views

When a masking view is deleted, all groups in the masking view remain intact. Any device reservations will continue to be enforced when a masking view is deleted.

To delete a masking view, use the following form:

```
symaccess -sid SymmID delete view -name ViewName
```

For example, to delete masking view `MV_1`, on Symmetrix 458, enter:

```
symaccess -sid 458 delete view -name mv_1
```

## Naming groups and views

Names for storage groups, port groups, initiator groups, and masking views can be up to 64 characters in length. Names are not case sensitive. They must be unique per group *type*. In other words, you can have the same name for a storage group, port group, and initiator group, but you cannot use the same name for two storage groups within the same array.

### Renaming

Use the `rename -new_name` command to rename a storage group, port group, or an initiator group, using the following form:

```
symaccess -sid SymmID rename -name GroupName -type <storage | port | initiator>
-new_name NewGroupName
```

Use the `rename view -new_name` command to rename a masking view, using the following form:

```
symaccess -sid SymmID rename view -name ViewName -new_name NewViewName
```

If the new name already exists, an error returns.

## Cascaded initiator groups

The renaming of an initiator group will be propagated to the higher group if the group is cascaded.

## Backing up and restoring views

The masking views, including storage groups, port groups, and initiator groups can be backed up to a file, and restored from the backup file.

The `backup` and `restore` functionality for `symaccess` provides compact data records. To maintain backward compatibility, the `backup` and `restore` commands can be used on files created by versions of Solutions Enabler lower than V7.3, however, the new format cannot be read or restored by these earlier versions of Solutions Enabler.

As a result of a restore, if a storage group violates FAST rules (such as, one device appears in multiple FAST controlled storage groups or a thin device gets restored to a storage group under FAST control), the restore will fail by default. The storage groups that cause the failure will be listed in the SYMAPI log file.

If the user specifies the `-disassociate` flag on the `restore` command, the storage groups causing a violation of FAST policies will be automatically disassociated from their FAST relationship. If the restore of a storage group from a backup file causes the violation, then the restored group's FAST association will be broken. All associations that are broken and the storage groups and devices that caused this break will be recorded in the SYMAPI log file. See [Chapter 4, “Fully Automated Storage Tiering,”](#) for details about using FAST.

## Unused storage groups

Use the `-unused_sgs` option with the `restore` command to put back stand-alone storage groups as well as those currently in a masking view.

[Table 14](#) lists behavior of the `restore` action when a combination of masking (Auto-provisioning), FAST, and unused storage groups are present in the backup file and may (or may not) have changed in the current Symmetrix configuration.

**Table 14** `symaccess` restore behavior (page 1 of 2)

Condition of the storage group in the backup file	Current condition of the storage group on the Symmetrix array	Default behavior (and condition of storage group after the restore)	Behavior when adding the <code>-unused_sgs</code> flag
Masking only	Masking only	The masking view will be restored from the backup file.	N/A
Masking or Masking + FAST Devices 19, 20, 21 are in the storage group.	Masking and FAST Devices 2, 3, 4 are in the storage group.	The storage group will have devices 19,20,21. The storage group will have the same FAST association that it had before the restore. FAST will be acting on devices 19,20,21 after the restore.	N/A
FAST only	Masking only or Masking and FAST	The storage group stays the way it is currently. However, masking is removed from this group. This means, that the views to which the group is associated right now, are dissolved.	N/A
FAST only or SG does not exist in the backup file.	Unused	The unused group is left on the Symmetrix array as is. The group is not restored from the backup file.	The group is removed from the array.

**Table 14** symaccess restore behavior (page 2 of 2)

Condition of the storage group in the backup file	Current condition of the storage group on the Symmetrix array	Default behavior (and condition of storage group after the restore)	Behavior when adding the <code>-unused_sgs</code> flag
Unused Devices 19, 20, 21 are in the storage group.	Masking Devices 2, 3, 4 are in the storage group.	The storage group stays the way it is currently (devices 2, 3, 4 in the SG). However, masking is removed from this group. This means, that the views to which the group is associated right now, are dissolved.	Masking is removed from this group. This means, that the views to which the group is associated right now, are dissolved. The group is restored from the backup file (devices 19, 20, 21 in the SG now).
Unused Devices 19, 20, 21 are in the storage group.	Unused or FAST Devices 2, 3, 4 are in the storage group.	The storage group stays the way it is currently (devices 2, 3 and 4 in the SG).	The group is restored from the backup file. If the group was under FAST control before the restore operation, the FAST association is preserved. FAST now acts on the restored group, that is, on devices 19,20,21.
Unused Devices 19, 20, 21 are in the storage group.	Masking and FAST Devices 2, 3, 4 are in the storage group.	Masking is removed from this group. This means, that the views that the group is associated with right now, are dissolved. In this case, devices 19,20,21 are no longer masked. The contents of the group remain as devices 2,3,4. FAST still acts on these devices.	Masking is removed from this group. This means, that the views to which the group is associated right now, are dissolved. In this case, devices 19,20,21 are no longer masked. The group is restored from the backup file and therefore has devices 19, 20, 21. The FAST association of the group is preserved and FAST now acts on these new devices.

Use the following syntax to backup the masking views for a Symmetrix array to a file:

```
symaccess -sid SymmID -f BackupFilename [-noprompt]
    backup
    restore [-remove_ckd] [-unused_sgs] [-disassociate]
```

The `symaccess` command will validate the consistency of the Auto-provisioning data before the `backup` or `restore` actions are performed.

Use the `-noprompt` option to eliminate the prompt for confirmation of the operation. Use the `-remove_ckd` option to skip all CKD devices within the backup, allowing the backup to be restored if the CKD devices are no longer mapped.

The `-disassociate` option disassociates the storage group from a FAST policy if the storage group contains invalid devices for FAST.

## Cascaded SGs

When restoring a backup file with cascaded SGs to versions of Enginuity earlier than 5876 parent SGs cannot be restored because they are not supported. In an effort to manage this issue, the command will operate as follows:

- ◆ All child SGs that are part of a view will be restored.
- ◆ For parent SGs that are in a view, the API will replace the content of the SG with the aggregate device list of its child SGs and will restore the converted SG. This restores an SG of devices that can be used to restore the view and allow devices in these views to maintain their masking after the restore.

The command will operate as follows when the `-unused_sgs` option is given:

- ◆ All unused child SGs will be restored.
- ◆ For parent SGs that are not part of a view, the API will replace the content of the SG with the aggregate device list of its child SGs and will restore the converted SG. Since the aggregate device list of a parent SG that is not in a view can exceed the maximum of 4096 devices device that can be added to a non-parent, some parent SGs may not be able to be converted. The API will verify that all the unused parent SGs can be converted and the restore operation will fail if one can't be converted.
- ◆ If one of the unused parent SGs cannot be converted and the `-force` option is given, Solutions Enabler will continue the `restore` converting all the SGs that can be converted. All parent SGs that cannot be converted will be empty SGs.

*Examples* The following example shows the message displayed when consistency errors are found during a backup operation:

```
symaccess backup -sid 266 -f aclx_backup -nop
```

```
Starting a backup operation.....
```

```
There are inconsistencies in the masking database. The operation cannot be
performed.
```

The following is an example of the displayed message when no consistency errors are found during the backup operation.

```
symaccess backup -sid 266 -f aclx_backup -nop
```

```
Starting a backup operation.....
```

```
The masking data on Symmetrix 000192600266 was backed up to file aclx_backup.
```

## Copying groups and views

The `symaccess copy` command provides the ability to copy a storage, port, or initiator group, or a complete masking view from one Symmetrix array to another Symmetrix array. When copying, any child view or cascaded initiator group will be included in the copy action.

Both arrays must be running Enginuity 5874 or higher.

To perform the copy action, use the following form:

```
symaccess -sid SymmID -target_sid SymmID

copy -name GroupName -type storage
    [-reserve_id ResvID[, ResvID[, ResvID...]]]

copy -name GroupName -type initiator | port

copy -name ViewName view [-ckd] [-celerra] [-rp]
    [-reserve_id ResvID[, ResvID[, ResvID...]]]
```

*Example* To copy masking view `mv_1` from Symmetrix 207 to Symmetrix 123, enter:

```
symaccess -sid 207 -target_sid 123 copy -name mv_1 view
```

Use the `-reserve_id` option to include any device reservations, use the `-ckd` option to specify that the view contains CKD devices, use the `-celerra` option to specify that the view contains Celerra devices (the devices will also be mapped), and use the `-rp` option to include devices that have been tagged for RecoverPoint.

## Managing storage groups

After creating a storage group, as explained in [“Creating storage groups” on page 138](#), devices can be added to and removed from the storage group. Other actions on a storage group include renaming the group and deleting the group. This section explains how to perform all these actions using the `symaccess` command.

### Adding devices

A storage group can contain up to 4k Symmetrix device numbers, and devices can belong to more than one storage group. When adding devices, specify the device names, a range of devices, a list of devices in a device group, or devices in a device file.

To add devices to an existing storage group, use the following form:

```
symaccess -sid SymmID -name GroupName -type storage
    [-reserve_id ResvID[, ResvID[, ResvID...]]]
    [-ckd] [-celerra] [-rp]

add devs SymDevStart:SymDevEnd [-lun Addr] | SymDevName [-lun Addr] |
    SymDevName, SymDevName, SymDevName...
    [-lun Addr | -lun Addr, Addr, Addr...]

add -g DgName [-std] [-bcv] [-vdev] [-tgt] [-lun Addr]
add -file DeviceFileName [src] [tgt] [-lun Addr]
add sg SgName [, SgName1, SgName2, ., SgNameN]
    [-lun Addr]
```

Device reservations will continue to be enforced when they are added to a storage group. Use the `-ckd` option for adding CKD devices to a storage group. Use the `-celerra` option for adding (and mapping) Celerra devices to a storage group. Use the `-rp` option for adding devices tagged for RecoverPoint to a storage group.

---

**Note:** Engenuity 5876 blocks adding AS/400 and RecoverPoint-tagged devices to a storage group that is in a masking view containing FCoE directors. In addition, a `restore` of AS/400 and RecoverPoint-tagged devices to a storage group that is in a masking view containing FCoE directors is not supported. When detected, the AS/400 and RecoverPoint-tagged devices will be removed prior to completing the `restore` action.

---

## Designating the LUN address

When devices are added at the storage group creation time, do not specify a LUN address. The LUN address is determined when the masking view is created.

LUN addresses should only be supplied if the storage group is already contained within a view. In this case, a single LUN can be given, or one for each device range. If the LUN address is not specified, the Symmetrix array will assign the LUN address.

## Cascaded SG restrictions

The following restrictions exist for child SGs belonging to a single parent SG:

- ◆ Only a single level of cascading is permitted. A parent SG may not be a child of another SG.
- ◆ SGs can only contain devices or other SGs. No mixing is permitted. This covers attempts to perform device add operations to a parent SG. This also covers attempts to add child SGs to an SG containing devices.
- ◆ A parent can have up to 32 child SGs.
- ◆ A parent SG cannot inherit the same device from more than one child SG.
- ◆ A child SG may only be contained by a single parent SG.
- ◆ An SG associated with a FAST policy is not allowed to be a parent SG.

In addition, the following existing restrictions affect `add` operations:

- ◆ When in a view the total number of devices inherited by a parent SG cannot exceed 4096 devices.
- ◆ If adding Celerra devices into an SG, or a child SG with Celerra devices to a parent SG within a view, the `-celerra` flag must be used.
- ◆ If adding a CKD device or a child SG with CKD devices to a parent SG within a view, the `-ckd` flag must be used.
- ◆ If adding a RecoverPoint (RP) tagged device or a child SG with RP tagged devices to a parent SG within a view, the `-rp` flag must be used.
- ◆ A device may not be added to an SG associated with a FAST policy if the device already exists in another SG that is also associated with a FAST policy.

The next section provides examples of adding devices to a storage group.

*Examples* To add devices 089, 090, and 091 to storage group `SG_Prod` on Symmetrix array 458, enter:

```
symaccess -sid 458 -name SG_Prod -type storage add devs 089,090,091
```

To add the BCV devices in device group `Prog2` to storage group `SG_Prod` on Symmetrix array 458, enter:

```
symaccess -sid 458 -name SG_Prod -type storage add -g Prog2 -bcv
```

To add the target devices in a device file `ProdFile` to storage group `SG_Prod` on Symmetrix array 458, enter:

```
symaccess -sid 458 -name SG_Prod -type storage add -file ProdFile tgt
```

Where `ProdFile` contains:

```
011      031
012      033
051      076
094      077
```

Devices 031, 033, 076, and 077 are added to `SG_Prod`.

## Setting Host I/O Limits

With Solutions Enabler V7.5, you can set Host I/O limits on the amount of Front-end (FE) bandwidth and I/Os per second (IOPs) that can be consumed by a set of Symmetrix devices over a set of director ports. The bandwidth and I/Os limits are then monitored by the Symmetrix to ensure that they do not exceed the specified maximum bandwidth or maximum IOPs. This feature allows you to place limits on the FE bandwidth and IOPs consumed by applications on the Symmetrix array.

---

**Note:** Host I/O Limits can only be set on Fibre Channel or iSCSI; FCOE is not supported.

---

The `symmsg` provides two options for setting and modifying the Host I/O limits for a storage group, as follows:

```
symmsg -sid SymmID [-i Interval] [-c Count] [-v]  
  set <[-bw_max <MBperSec | NOLIMIT>]  
    [-iops_max <IOperSec | NOLIMIT>]>
```

Where:

`-bw_max` — Specifies the front-end maximum bandwidth in MBs/sec for the storage group. The maximum bandwidth is set to unlimited if `NOLIMIT` is specified.

`-iops_max` — Specifies the front-end maximum IOs/sec. The maximum IOs/sec is set to unlimited if `NOLIMIT` is specified.

To configure the Host I/O limits for a set of devices, the limit is set for a storage group. When you create a provisioning view using that storage group, the limit is applied to the devices in the storage group for the ports defined in the port group.

The Host I/O Limits can be added, removed, and modified for a storage group. For a cascaded storage group, the Host I/O Limit can be added for either the parent or the child storage group, but not for both. If a parent storage group has limits set, those settings are

shared among all its child storage groups when a provisioning view is created using the parent storage group. When a parent storage group has limits set, you cannot create provisioning views using the child storage groups.

Refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide* for all the feature details and restrictions.

## Removing devices

A storage group should not be deleted until all the devices have been removed. To remove devices from a storage group, or child storage groups, use the following form:

```
symaccess -sid SymmID -name GroupName -type storage
  [-reserve_id ResvID[, ResvID[, ResvID...]]] [-force]
  [-unmap [-celerra]] [-ckd] [-rp]

remove devs SymDevStart:SymDevEnd | SymDevName |
  SymDevName, SymDevName, SymDevName...

remove -g DgName [-std] [-bcv] [-vdev] [-tgt]

remove -file DeviceFileName [src] [tgt]

remove sg SgName[, SgName1, SgName2, .. SgNamen]
```

## Cascaded SG restrictions

The following restrictions exist for child and parent SG:

- ◆ If the parent SG has a masking view, removing devices from the child SGs will not be permitted if it causes the parent SG to have no more devices.
- ◆ If the parent SG has a masking view, removing a child SG is not permitted if it causes the parent SG to inherit no devices.
- ◆ Devices cannot be removed from a child SG using the parent SGs name.

In addition, the following existing restrictions affect remove operations:

- ◆ If removing Celerra devices from an SG or a child SG with Celerra devices from a parent SG within a view, the `-celerra` flag must be used.
- ◆ If removing a CKD device or a child SG with CKD devices from a parent SG within a view, the `-ckd` flag must be used.
- ◆ If removing an RP tagged device or a child SG with RP tagged devices from a parent SG within a view, the `-rp` flag must be used.

**Example** To remove the BCV devices in device group `Prog2` from storage group `SG_Prod` on Symmetrix array 458, enter:

```
symaccess -sid 458 -name SG_Prod -type storage remove -g Prog2 -bcv
```

---

**Note:** A storage group cannot be completely emptied if it is associated with a masking view or in use by FAST.

---



## Deleting storage groups

A storage group should be empty before being deleted, unless using the `-force` option.

```
symaccess -sid SymmID

delete <view -name ViewName [-unmap [-emulation celerra]]
      [-reserve_id ResvID[,ResvID[,ResvID...]]] > |
< -name <GroupName> -type <storage
      [-reserve_id ResvID[,ResvID[,ResvID...]]] |
port | initiator> [-force] > [-noprompt][-emulation ckd]
```

For example, to delete storage group `SG_1` from Symmetrix 458, enter:

```
symaccess -sid 458 -name sg_1 -type storage delete
```

---

**Note:** A storage group cannot be deleted if it is associated with a masking view or is in use by FAST. To delete both a storage group and a masking view, delete the masking view first, then delete the storage group.

---

Use the `-unmap` option to unmap the devices from the ports contained in the port group of the view. If a device is mapped to the same port through another view, it will not be unmapped until the second masking view is deleted.

## Managing port groups

Port groups contain director and port identification and belong to a masking view. Ports can be added and removed. When a port group is no longer associated with a masking view, it can be deleted.

In addition, CHAP authentication can be enabled and disabled on port groups. This section explains the following port-related tasks:

[“Adding ports” on page 154](#)

[“Removing ports” on page 154](#)

[“Deleting port groups” on page 154](#)

[“Locking down a Fibre Channel ID” on page 155](#)

### Adding ports

Ports can be added to an existing port group by specifying the name of the group and indicating the group type and the director port combinations, using the following form:

```
symaccess -sid SymmID -name GroupName -type port
           -dirport Dir:Port[,Dir:Port[,Dir:Port...]] [-ckd]

add [-celerra] [-rp]
```

To add port 1 of Fibre director 16D to the existing port group PG\_1 on Symmetrix 245, enter:

```
symaccess -sid 245 -name PG_1 -type port add -dirport 16D:1
```

### Removing ports

To remove a port, use the following form:

```
symaccess -sid SymmID -name GroupName -type port
           -dirport Dir:Port[,Dir:Port[,Dir:Port...]] [-ckd] [-unmap] [-force]

remove [-unmap [-celerra] [-rp]]
```

To remove port 1 of Fibre director 16D from the existing port group PG\_1 on Symmetrix 245, enter:

```
symaccess -sid 245 -name PG_1 -type port remove -dirport 16D:1
```

---

**Note:** A port group cannot be completely emptied if it is associated with a masking view.

Use the `-unmap` option to unmap all devices in the storage group associated with the view. Use the `-force` option to remove the port.

### Deleting port groups

To delete a port group, use the following form:

```
delete <view -name ViewName> [-force] |
      -name GroupName -type <storage | port | initiator > [-noprompt]
```

---

**Note:** A port group cannot be deleted if it is associated with a masking view.

To delete port group PG\_1 on Symmetrix 245, enter:

```
symaccess -sid -name PG_1 -type port delete
```

## Locking down a Fibre Channel ID

Fibre Channel ID (FCID) lockdown is a security feature that limits host device access by adding Fibre Channel ID information of a switch within a fabric to device access records in the login history table. This feature handles WWN spoofing and the threat it poses to your networked systems in a shared (same director port) storage port configuration.

For example, to implement the Fibre Channel ID lockdown feature on Fibre Channel 021300 for director 16A, port 0, enter:

```
symaccess -sid 018 set lockdown on 021300
```

This feature lets you set the Fibre Channel ID (FCID) of the WWN of the HBA you want to protect. The FCID is then added to the database record for the WWN of the specified HBA with the specified director and is locked. Once a Fibre Channel ID is locked, no user with a spoofed WWN can log in. If a user with a spoofed WWN is already logged in, that user loses all access through that HBA.

### CAUTION

When an HBA logs in to a director port, the Fibre Channel ID accompanies it, telling the director port where to send its response. By specifying Fibre Channel ID information of the switch, the valid physical path through the SAN for a particular HBA is locked down. Only an HBA with a Fibre Channel ID that matches the FCID specified in the device masking record is able to log in to the storage port. It is recommended that at least two HBAs be available on the administrator host. If one HBA becomes locked out, the host will have access through the other HBA and can correct the record in the database.

## Lockdown steps

To find the Fibre Channel ID, lock it down, verify that it is locked down, and then force the change to take effect, use the following procedure:

1. Find the WWN. If the device is visible, run `symaccess list hba` to find the device path of the HBA you want to protect.
2. Find the Fibre Channel ID value by referring to [“Finding the FCID of a switch” on page 350](#).
3. Run `symaccess set lockdown` to on with the FCID of the Fibre Channel ID you found in step 2.
4. To finalize the change, either reboot the host or pull the cable from the director and then replace the cable.

## Managing initiator groups

This section explains how to manage initiator groups. The following topics are included:

[“Adding initiators” on page 156](#)

[“Removing initiators” on page 157](#)

[“Deleting initiator groups” on page 157](#)

[“Setting initiator group flags” on page 157](#)

[“Setting HBA flags” on page 159](#)

[“Replacing an HBA” on page 160](#)

[“Renaming an HBA” on page 161](#)

[“Using CHAP authentication” on page 161](#)

## Adding initiators

Initiators can be added to an existing initiator group by specifying the initiator type (`-wwn` or `-iscsi`), the initiator group name, or by using an input file. The following is the syntax for adding one or more initiators to an initiator group:

```
symaccess -sid SymmID -name GroupName -type initiator
           -wwn wwn | -iscsi iscsi | -ig InitiatorGroupName |
           -f InitiatorFilename
add
```

### Individual initiators

For example, to add initiator `-wwn 10000000c94ef69c` to the initiator group `IG_1` on Symmetrix 245, enter:

```
symaccess -sid 245 -name IG_1 add -type initiator -wwn 10000000c94ef69c
```

### File

When using an input file, each initiator must be placed on a new line and start with either `WWN:` or `iSCSI:` or `IG:`, depending on the type of the initiator or initiator group name. The following is an example of the format for an initiator file:

```
WWN:10000000c94ef69c
iSCSI:iscsiname
IG:IgName
#WWN:10000000c94ef69d
```

If the format of the initiator does not match the label at the start of the line, the file return an error. A commented line, which the system ignores, is specified by placing the pound sign (#) at the beginning of a line.

## Cascaded initiator groups

An initiator group can be added to another initiator group, only if it does not contain any initiator groups.

The following scenario describes cascaded initiator groups:

- ◆ HOST1 contains WWN1 & WWN2, which are added to IG\_1.

- ◆ HOST2 contains WWN3 & WWN4, which are added to IG\_2.
- ◆ IG\_3 is created and contains IG\_1 & IG\_2.

In this example, gatekeeper devices for HOST1 can be assigned to IG\_1, while different gatekeeper devices for HOST2 can be assigned to IG\_2. The application devices needed by both hosts can be assigned to IG\_3.

---

**Note:** If using the Volume Set Addressing flag, both the parent and child initiator group must have the flag.

---

## Removing initiators

The `symaccess remove` command removes specified initiators from an initiator group.

An initiator group that is currently associated with a masking view will be allowed to be emptied and the view will remain along with the other two groups. The following is the syntax for removing an initiator from an initiator group:

```
symaccess -sid SymmID -name GroupName -type initiator
           -wwn wwn | -iscsi iscsi | -ig InitiatorGroupName
           | -f InitiatorFilename [-login]
remove
```

For example, to remove initiator `-wwn 10000000c94ef69c` from the initiator group `IG_1` on Symmetrix 245, enter:

```
symaccess -sid 245 -name IG_1 remove -type initiator -wwn 10000000c94ef69c
```

Add the `-login` option to the command to remove the initiator from the Symmetrix array's login history table.

## Deleting initiator groups

To delete an initiator group, use the following form:

```
delete <view -name ViewName> [-force] |
      -name GroupName -type <storage | port | initiator > [-noprompt]
```

To delete initiator group `IG_1` on Symmetrix 245, enter:

```
symaccess -sid -name IG_1 -type initiator delete
```

---

**Note:** An initiator group cannot be deleted if it is associated with a masking view.

---

## Setting initiator group flags

This feature allows you to set all the available port flags at the initiator group level. The following flags are supported for initiator groups:

Volume_Set_Addresssing	[V]
Common_Serial_Number	[C]
Disable_Q_Reset_on_UA	[D]
Environ_Set	[E]
Avoid_Reset_Broadcast	[ARB]
AS400	[AS4]
OpenVMS	[OVMS]
SCSI_3	[SC3]
SPC2_Protocol_Version	[SPC2]
SCSI_Support1	[OS2007]

---

**Note:** A flag cannot be set for the group if it conflicts with any initiator in the group. After a flag is set for a group, it cannot be changed on an initiator basis.

---

To set an override flag for an initiator group, use the following form:

```
symaccess -sid SymmID -name GroupName -type initiator
        set ig_flags <on <Flag> <-enable |-disable> | off [Flag]>
```

Where:

**on** — Turns on the specified initiator group port *Flag* override and allows you to set the status of that flag to either **enabled** or **disabled**.

**off** — Turns off the initiator group port *Flag* override.

**enable** — Sets the status of the initiator group port *Flag* to **enabled**. The status can only be set when the initiator group port flag override setting value is **on**.

**disable** — Sets the status of the initiator group port *Flag* to **disabled**. The status can only be set when the initiator group port flag override setting value is **on**.

**Example** To set the OS2009 [OS2009] for the initiator group *my\_ig* on Symmetrix 266, enter:

```
symaccess -sid 266 -type init -name my_ig set ig_flags on OS2009 -enable
```

Use the **show** command with the **-detail** option to view the flag from the previous command:

```
symaccess -sid 266 show my_ig -type init -detail
```

Output similar to the following example displays:

```
Symmetrix ID           : 000192600266

Initiator Group Name   : my_ig
Last updated at       : 10:52:15 AM on Wed Mar 31, 2010

Port Flag Overrides    : Yes
  Enabled              : OS2009 (OS2009)
                        Common_Serial_Number (C)
  Disabled             : Avoid_Reset_Broadcast (ARB)
Consistent Lun         : No

Originator Port wwn    : 1234567822446688
User-generated Name    : 1234567822446688/1234567822446688
FCID Lockdown          : No
Heterogeneous Host     : No
Port Flag Overrides    : Yes
  Enabled              : OS2009 (OS2009)
                        Common_Serial_Number (C)
  Disabled             : Avoid_Reset_Broadcast (ARB)
CHAP Enabled           : N/A
Type                   : Fibre
```

---

**Note:** The Write Protect Bypass flag is only supported on Symmetrix arrays running Enginuity 5875. The flag is not supported in Enginuity 5876. If a backup file contains an initiator group with the Write Protect Bypass flag set, a **restore** operation on a Symmetrix array running Enginuity 5876 will clear the flag from the initiator group and all its initiators.

---

## Setting HBA flags

Symmetrix arrays running Solutions Enabler V7.0 and Enginuity 5874 and higher, only allow you to set the HBA port flags on a per initiator basis. The HBA must belong to an initiator group. This feature allows specific host flags to be enabled and disabled on the director port.

**Note:** Setting HBA port flags replaces setting the heterogeneous host configuration flags. To switch to setting HBA port flags, the heterogeneous host configuration must be disabled for a given HBA and all flags must be reset.

To set (or reset) the HBA flags, use the following form:

```
symaccess -sid SymmID -wwn wwn | -iscsi iscsi

set hba_flags <on <flag,flag,flag...> <-enable |-disable> |
    off [flag,flag,flag...]>
```

Where:

**hba\_flags** — Sets the record in the database to hold information on the HBA port setting that may differ than the current setting on the FA.

**on | off** — Turns HBA flags on or off.

**flag** — Specifies the overridden HBA port flags from the values [in brackets]:

Supported HBA port flags		Supported initiator group port flags	
AS400	[AS4]	AS400	[AS4]
Avoid_Reset_Broadcast	[ARB]	Avoid_Reset_Broadcast	[ARB]
Common_Serial_Number	[C]	Common_Serial_Number	[C]
Disable_Q_Reset_on_UA]	[D]	Disable_Q_Reset_on_UA	[D]
Environ_Set	[E]	Environ_Set	[E]
OpenVMS	[OVMS]	OpenVMS	[OVMS]
SCSI_3	[SC3]	SCSI_3	[SC3]
SCSI_Support1	[OS2007]	SCSI_Support1	[OS2007]
SPC2_Protocol_Version	[SPC2]	SPC2_Protocol_Version	[SPC2]
		Volume_Set_Addresssing]	[V]

**-enable** — Enables the specified HBA port flag(s) on a per initiator basis.

**-disable** — Disables the specified HBA port flag(s) on a per initiator basis.

For example, to turn on HBA flags and enable the `Common_Serial_Number` and `SCSI_3` flags, and disable the `Disable_Q_Reset_on_UA` flag, on an HBA with the WWN ID `210000e08b0995b7` for Symmetrix 031, director 16A port 0, enter:

```
symaccess -sid 031 set hba_flags on C,SC3 -enable -wwn 210000e08b0995b7-dir 16A -p 0
symaccess -sid 031 set hba_flags on D -disable -wwn 210000e08b0995b7-dir 16A -p 0
```

The `symaccess show -detail` output has been modified to display the flags that are turned on and off for each HBA initiator that has this feature enabled, as shown in the following example:

```
symaccess -sid 237 -type initiator -detail show Prod1

Symmetrix ID           : 000190300237
Last updated at        : 08:46:54 AM on Tue Jul 29,2008

Initiator Group Name    : Prod1
Originator Port wwn     : 10000000c94ef69c
  User-generated Name   : api196/10000000c94ef69c
  FCID Lockdown         : No
  Heterogeneous Host    : No
  Port Flag Overrides   : No
  Type                  : Fibre

Originator Port wwn     : 5006016839a00c5c
  User-generated Name   : 5006016839a00c5c/5006016839a00c5c
  FCID Lockdown         : No
  Heterogeneous Host    : No
  Port Flag Overrides   : No
  Type                  : Fibre

iSCSI Name             : Symm_iScsi
  User-generated Name   : iScsi_node_alias/iScsi_port_alias
  FCID Lockdown         : N/A
  Heterogeneous Host    : No
  Port Flag Overrides   : No
  Type                  : iScsi

Group Name              : IniGrp
  User-generated Name   : N/A
  FCID Lockdown         : N/A
  Heterogeneous Host    : N/A
  Port Flag Overrides   : N/A
  Type                  : Initiator Group
```

## Replacing an HBA

If a host adapter fails, or needs replacement for any reason, you can replace the adapter and assign its set of devices to a new adapter by using the `replace` action in the following form:

```
symaccess replace -wwn wwn -new_wwn NewWWN [-noprompt]

symaccess replace -iscsi iscsi -new_iscsi NewiSCSI [-noprompt]
```

To swap HBAs:

1. Run `symaccess list logins` to view the old WWN/iSCSI HBAs.
2. Swap the HBA boards.
3. Run `symaccess list hba` or `discover` to view the new initiator (for example WWN).
4. Run `symaccess replace` to substitute a new WWN for all occurrences of the old WWN. For example, to replace old WWN 20000000c920b484 with new WWN 20000000c920b393:

```
symaccess -sid 814 replace -wwn 20000000c920b484 -new_wwn 20000000c920b393
```

5. Run `symaccess discover -rename` to establish the new AWWN and assign an AWWN to the new HBA in the login history table.



## Renaming an HBA

To rename the alias for a specified initiator within a group and the login history table, use the following form:

```
symaccess -sid SymmID
          rename -wwn wwn | -iscsi iscsi -alias alias
```

## Using CHAP authentication

CHAP (Challenge Handshake Authentication Protocol) allows you to manage a credential name and a CHAP secret, which are similar to a username and a password, though more secure than the standard Password Authentication Procedure (PAP).

To enable CHAP on an iSCSI initiator, use the following form:

```
symaccess -sid SymmID -iscsi iscsi enable chap
```

To set the CHAP credential and secret on a director and port, use the following form:

```
symaccess -sid SymmID -dirport Dir:Port
          set chap -cred Credential -secret Secret
```

To enable CHAP on a specific director and port, use the following form:

```
symaccess -sid SymmID [-dirport Dir:Port] enable chap
```

To disable CHAP on a specific director and port, use the following form:

```
symaccess -sid SymmID [-dirport Dir:Port] disable chap
```

To delete CHAP from a specific director and port, use the following form:

```
symaccess -sid SymmID [-dirport Dir:Port] delete chap
```

## Displaying Auto-provisioning Group information

With Solutions Enabler V7.2, when you list information without using the `-detail` option, any column without data will not display. If the `-detail` option is provided, the column without data will display a dash (-).

This section shows the output for the masking views using the following form:

```
symaccess -sid SymmID | -file BackupFilename

list [-name GroupName] [-v]

list -type <storage [-devs <SymDevName[:SymDevName]>] | port
      [-dirport Dir:Port] | initiator [-wwn wwn |
      -iscsi iscsi]> [-name GroupName] [-detail | -v]

list devinfo [-ig InitiatorGroupName]

list view [-name ViewName][-v][-detail]

list chap [-dirport Dir:Port][-v]

show GroupName -type <initiator [-detail] | port | storage>

show view ViewName [-ig ChildInitiatorGroupName]

symaccess -sid SymmID

list assignment [-v] -devs <SymDevStart:SymDevEnd |
      SymDevName | <SymDevName,SymDevName...>

list no_assignments [-dirport Dir:Port]
```

### Showing the last update time

With Solutions Enabler V.5 and higher, the displays for `symaccess` return the current Symmetrix time and the time when the group was modified in seconds. This delta time reports the last updated time of a provisioning group, or any related groups of the selected group.

## Displaying masking views

The following example shows the command and output for listing all masking views on Symmetrix array 237:

```
symaccess -sid 237 list view
```

```
Symmetrix ID           : 000190300237
```

Masking View Name	Initiator Group	Port Group	Storage Group
View1	IG_1	PG_1	SG_1
View2	WinHost	PG_5	Acct
...			

The `symaccess show view` command displays the last update time of the masking view. The groups that make up a masking view include the port group, the initiator group and any child initiator groups, the storage group and any child storage groups:

```
symaccess show view HR_view
```

```
Symmetrix ID           : 000195700601
```

```

Masking View Name           : HR_view
Last update time           : 06:43:59 AM on Wed Mar 21,2012
View last update time      : 08:44:00 AM on Thu Mar 22,2012

Initiator Group Name       : HR_hosts

Host Initiators
{
  WWN   : 210000e08b80e873 [alias: DLDV0191/210000e08b80e873]
  WWN   : 210100e08ba0e873 [alias: DLDV0191/210100e08ba0e873]
  Ig    : HR_host_GrpA
}

Port Group Name            : HR_ports

Director Identification
{
  FA-15F:1
}

Storage Group Name         : HR_storage

Number of Storage Groups   : 2
Storage Group Names        : HR_Stor_DeptA (IsChild)
                           : HR_Stor_DeptB (IsChild)

Sym      Dir:P  Physical Device Name      Host  Attr  Cap(MB)
Dev      -----
-----
1A41     15F:1  Not Visible                  0      60000
1A42     15F:1  Not Visible                  1      60000
1A43     15F:1  Not Visible                  2      60000
1A44     15F:1  Not Visible                  3      60000
1A45     15F:1  Not Visible                  4      60000
-----
Total Capacity              300000

```

Use the `-detail` option with the `list` command to display all the details of a view, including the child initiator groups.

## Viewing group details

Storage group, port group, and initiator group details can be displayed using the `symaccess list` and `symaccess show` commands, as follows:

```

symaccess -sid SymmID | -f BackupFilename

list [-name GroupName] [-v]

list -type <storage [-devs SymDevName <:SymDevName>] | port
      [-dirport Dir:Port] | initiator [-wwn wwn |
      -iscsi iscsi]> [-v] [-name GroupName]

show GroupName -type <storage | port | initiator [-detail]>
show view ViewName

```

An optional group name may be supplied as input and only the details on that group return.

An optional director and port number may be supplied and only the port groups that belong to the director and port return.

To display a list of the initiator groups, in verbose mode, use the following form:

```
list -type initiator [-wwn wwn | -iscsi iscsi] -v
```

To display a specific initiator group, in verbose mode, use the following form:

```
list -name GroupName -v
```

To display the details of a specific initiator group, use the following form:

```
show GroupName -type initiator -detail
```

To display information about specific devices, or device ranges, use the following form:

```
symaccess -sid SymmID | -f BackupFilename
list -type <storage [-devs SymDevName <:SymDevName>]
[-name SgName] [-detail | -v]
```

The `symaccess list -type storage -v` command output display the time since the last update of the storage group or any child storage group:

```
symaccess list -type storage -v

Symmetrix ID           : 000195700601

Storage Group Name     : backup_storage
Device Count           : 1
Storage Group Count    : 0
Masking View Count     : 0
Last update time       : 12:32:26 AM on Fri Apr 06,2012
Group last update time : 12:32:26 AM on Fri Apr 06,2012
Masking View Names     : None

Storage Group Name     : Mkt_storage (IsParent)
Device Count           : 0
Storage Group Count    : 1
Masking View Count     : 0
Last update time       : 08:10:22 PM on Tue Feb 14,2012
Group last update time : 10:32:26 AM on Fri Apr 06,2012
Masking View Names     : Mkt_view *
. . .
* Denotes Masking Views through a cascaded group
```

The `symaccess list view -detail` command displays the last update time of the masking view or any of the groups that make up the masking view:

```
symaccess list view -detail

Symmetrix ID           : 000195700601

Masking View Name       : HR_view
Last update time       : 04:30:24 AM on Tue Apr 03,2012
View last update time  : 04:35:36 AM on Tue Apr 03,2012

Initiator Group Name    : HR_hosts

Host Initiators
{
  IG      : HR_host_GrpA
}

Port Group Name         : HR_ports

Director Identification
{
  FA-15E:1
```

```

    }

Storage Group Name          : HR_storage

Number of Storage Groups : 0
Storage Group Names       : None

Sym   Dir:P  Physical Device Name      Host   Attr  Cap(MB)
Dev   -----
-----
1C27   15E:1  Not Visible                1      300000
-----
Total Capacity              300000
. . .

```

## Viewing device assignments

To display the assignments for one or more devices, use the following form:

```

symaccess -sid SymmID list assignments
          -devs SymDevStart:SymDevEnd | SymDevName |
              SymDevName, SymDevName... [-v]

```

**Example** To list the assignments for device range 20:22 and device 24 on Symmetrix 120, enter:

```
symaccess -sid 120 list assignments -devs 20:22,24
```

```

Symmetrix ID                : 000192600120

Device  Identifier          Type  Dir:P
-----
0020    10000000c9594dce    FIBRE  FA-7E:1
        210000e08b04daac    FIBRE  FA-7E:1
        210000e08b1ed7f1    FIBRE  FA-7E:1
0021    10000000c9594dce    FIBRE  FA-7E:1
        210000e08b04daac    FIBRE  FA-7E:1
        210000e08b1ed7f1    FIBRE  FA-7E:1
0022    10000000c9594dce    FIBRE  FA-7E:1
        210000e08b04daac    FIBRE  FA-7E:1
        210000e08b1ed7f1    FIBRE  FA-7E:1
0024    10000000c9594dce    FIBRE  FA-7E:1
        210000e08b04daac    FIBRE  FA-7E:1
        210000e08b1ed7f1    FIBRE  FA-7E:1

```

## Listing no assignments

To display a list of devices with no assignments, use the following form:

```
symaccess -sid SymmID list no_assignments [-dirport DirNum:PortNum]
```

**Example** To list the devices without assignments for Symmetrix 120, enter:

```
symaccess -sid 120 list no_assignments
```

Output similar to the following example displays:

```

Symmetrix ID                : 000192600120

Director Identification     : FA-7F
Director Port               : 0
ACLX Enabled                : No

No devices were found for this director/port

Director Identification     : FA-7F

```

```
Director Port          : 1
ACLX Enabled          : Yes
```

```
Devices not yet assigned :
```

```
030
031
032
033
034
```

```
...
```

## Listing initiator group devices

To view all the devices masked to an initiator group, use the following form:

```
symaccess -sid SymmID list devinfo [-ig InitiatorGroupName]
```

Initiator group displays with Solutions Enabler V7.5 include the child initiator group device information when adding the -detail option.

**Example** To list the details for initiator group `cluster_hosts` on Symmetrix 601, enter:

```
symaccess -sid 601 list devinfo -ig cluster_hosts -detail
```

```
Symmetrix ID          : 000192600601
```

```
Initiator Group Name   : cluster_hosts
```

```
Last update time      : 10:08:58 AM on Wed Apr 11,2012
```

```
Group last update time : 10:08:58 AM on Wed Apr 11,2012
```

```
Host Initiators
```

```
{
  IG   : exchange_hosts
}
```

Sym Dev	Dir:P	Physical Device Name	Host Lun	Attr	Cap (MB)	Masking View Name
1C27	15E:1	Not Visible	1		300	cluster_view
1C28	15E:1	Not Visible	2		938	cluster_view
1C29	15E:1	Not Visible	3		938	cluster_view
1C2A	15E:1	Not Visible	4		938	cluster_view
1C2B	15E:1	Not Visible	5		938	cluster_view
1C2C	15E:1	Not Visible	6		938	cluster_view
1C77	15E:1	Not Visible	7		938	cluster_view
1C79	15E:1	Not Visible	8		938	cluster_view
1C7A	15E:1	Not Visible	9		938	cluster_view
1C7B	15E:1	Not Visible	a		938	cluster_view
1C7C	15E:1	Not Visible	b		938	cluster_view
1C7D	15E:1	Not Visible	c		938	cluster_view
1C7E	15E:1	Not Visible	d		938	cluster_view
1C7F	15E:1	Not Visible	e		938	cluster_view
1C80	15E:1	Not Visible	f		938	cluster_view
1C82	15E:1	Not Visible	10		938	cluster_view
1C83	15E:1	Not Visible	11		938	cluster_view
1C85	15E:1	Not Visible	12		938	cluster_view
1C86	15E:1	Not Visible	13		938	cluster_view
1C88	15E:1	Not Visible	14		938	cluster_view
1C89	15E:1	Not Visible	15		938	cluster_view
1C8A	15E:1	Not Visible	16		938	cluster_view
1C8B	15E:1	Not Visible	17		938	cluster_view

```
Total Capacity          20936
```

```
Initiator Group Name   : exchange_hosts
Last update time      : 10:08:58 AM on Wed Apr 11,2012
Group last update time : 04:19:06 AM on Tue Apr 07,1970
```

## Host Initiators

```
{
  WWN   : 210000e08b80e873 [alias: DLDV0191/210000e08b80e873]
}
```

Sym Dev	Dir:P	Physical Device Name	Host Lun	Attr	Cap (MB)	Masking View Name
1C27	15E:1	Not Visible	1		300	cluster_view
1C28	15E:1	Not Visible	2		938	cluster_view
1C29	15E:1	Not Visible	3		938	cluster_view
1C2A	15E:1	Not Visible	4		938	cluster_view
1C2B	15E:1	Not Visible	5		938	cluster_view
1C2C	15E:1	Not Visible	6		938	cluster_view
1C77	15E:1	Not Visible	7		938	cluster_view
1C79	15E:1	Not Visible	8		938	cluster_view
1C7A	15E:1	Not Visible	9		938	cluster_view
1C7B	15E:1	Not Visible	a		938	cluster_view
1C7C	15E:1	Not Visible	b		938	cluster_view
1C7D	15E:1	Not Visible	c		938	cluster_view
1C7E	15E:1	Not Visible	d		938	cluster_view
1C7F	15E:1	Not Visible	e		938	cluster_view
1C80	15E:1	Not Visible	f		938	cluster_view
1C82	15E:1	Not Visible	10		938	cluster_view
1C83	15E:1	Not Visible	11		938	cluster_view
1C85	15E:1	Not Visible	12		938	cluster_view
1C86	15E:1	Not Visible	13		938	cluster_view
1C88	15E:1	Not Visible	14		938	cluster_view
1C89	15E:1	Not Visible	15		938	cluster_view
1C8A	15E:1	Not Visible	16		938	cluster_view
1C8B	15E:1	Not Visible	17		938	cluster_view
					-----	
Total Capacity					20936	

## Viewing the HBA alias name

Solutions Enabler V7.3 supports displaying both the WWN and the HBA alias name, as shown in the following example command and output:

```
symaccess -sid 266 -type init show init_grp
```

```
Symmetrix ID           : 000192600266
```

```
Initiator Group Name   : init_grp
```

```
Last updated at       : 01:40:56 PM on Wed Jan 12,2011
```

## Host Initiators

```
{
  WWN   : 1177446622889944 [alias: 1177446622889944/1177446622889944]
}
```

## Masking View Names

```
{
  view1
  myview1
}
```

## Parent Initiator Groups

```
{
  None
}
```

}

## Auto-provisioning Groups examples

*Group Examples* To create an initiator group named `initexample`, and to add WWN initiator `210000e08b04daac` to it, enter:

```
symaccess -sid 234 -type initiator -wwn 210000e08b04daac -name initexample create
```

To create a storage group named `storeexample`, and to add device `0026` to it, enter:

```
symaccess -sid 234 -type storage devs 0026 -name storeexample create
```

To create a port group named `portexample`, and to add director `7E` and port `1` to it, enter:

```
symaccess -sid 234 -type port -dirport 7E:1 -name portexample create
```

To add devices to a storage group named `storeexample`, enter:

```
symaccess -sid 234 -type storage -name storeexample add devs 0027
```

To remove WWN initiator `210000e08b04daac` from initiator group named `initexample`, enter:

```
symaccess -sid 234 -type initiator -wwn 210000e08b04daac -name initexample remove
```

To delete an initiator group named `initexample`, enter:

```
symaccess -sid 234 -type initiator -name initexample delete
```

To list all initiator, port and storage groups, enter:

```
symaccess -sid 234 list
```

*View Examples* To create a view named `viewexample`, containing initiator group `initexample`, port group `portexample`, and storage group `storeexample`, enter:

```
symaccess -sid 234 -name viewexample -sg storeexample -pg portexample -ig initexample create view
```

To delete a view named `viewexample`, enter:

```
symaccess -sid 234 -name viewexample delete view
```

To rename a view named `viewexample` to `mvexample`, enter:

```
symaccess -sid 234 rename view -name viewexample -new_name mvexample
```

To create a view, storage, port and initiator group named `TEST`, and to add devices to storage group `TEST`, initiators to initiator group `TEST`, and director ports to port group `TEST`, enter:

```
symaccess -sid 234 create view -name TEST -wwn 210000e08b04daac -dirport 7E:1 devs 0026
```

*Backup and Restore Examples* To back up groups and views of Symmetrix ID 234 to a file `backup_from_lab`, enter:

```
symaccess -sid 234 backup -file backup_from_lab
```

To restore group and view information in file `backup_from_lab` to Symmetrix ID 234, enter:

```
symaccess -sid 234 restore -file backup_from_lab
```



# CHAPTER 4

## Fully Automated Storage Tiering

This chapter describes Fully Automated Storage Tiering (FAST) concepts and how to use the `symfast` and `symtier` commands of SYMCLI. This chapter covers the following topics:

◆ What is FAST? .....	170
◆ FAST overview .....	172
◆ FAST VP overview .....	175
◆ Managing Symmetrix tiers .....	178
◆ Managing policies .....	188
◆ Managing storage groups .....	193
◆ Managing the FAST controller .....	200
◆ Displaying FAST information .....	211
◆ FAST reports .....	214

## What is FAST?

Fully Automated Storage Tiering (FAST™) is Symmetrix software that runs background algorithms to continuously analyze the utilization (busy rate) of the Symmetrix array devices. The FAST controller processes the algorithm data, and generates plans for moving and swapping data volumes to fine tune performance and reduce costs. FAST can move the most-used data to the fastest (and most expensive) storage, such as Enterprise Flash Drives (EFD), the least-used data to the slowest (and least expensive) storage, such as SATA, while maintaining the remaining data on Fibre Channel (FC) drives, based on user-defined Symmetrix tiers and FAST policies. The objective of tiered storage is to minimize the cost of storage, while improving or maintaining performance, by putting the right data, on the right Symmetrix tier, at the right time.

Solutions Enabler V7.4 introduced Federated Tiered Storage (FTS), which allows the virtualization of external storage as an external disk (eDisk). Adding the eDisk to the Symmetrix array makes its capacity available to the array as an external spindle.

With Solutions Enabler V7.5 the FTS tier can be associated with a technology type. The technology associated with the FTS tier indicates to the FAST VP controller the expected performance from the tier. This feature enables you to place the FTS tier at the right location for the expected performance of the external tier.

Refer to [Chapter 5, “Federated Tiered Storage”](#) for the configuration details of FTS.

After configuration, FAST can be set to move/swap data automatically or with user approval. All three drive technologies (EFD, FC, and SATA), or external (eDisks), are not required in the Symmetrix array to use FAST; it can also work between two technologies.

There are two FAST products: FAST and FAST for virtual pools (FAST VP). The differences between these two versions are highlighted in [Table 15](#).

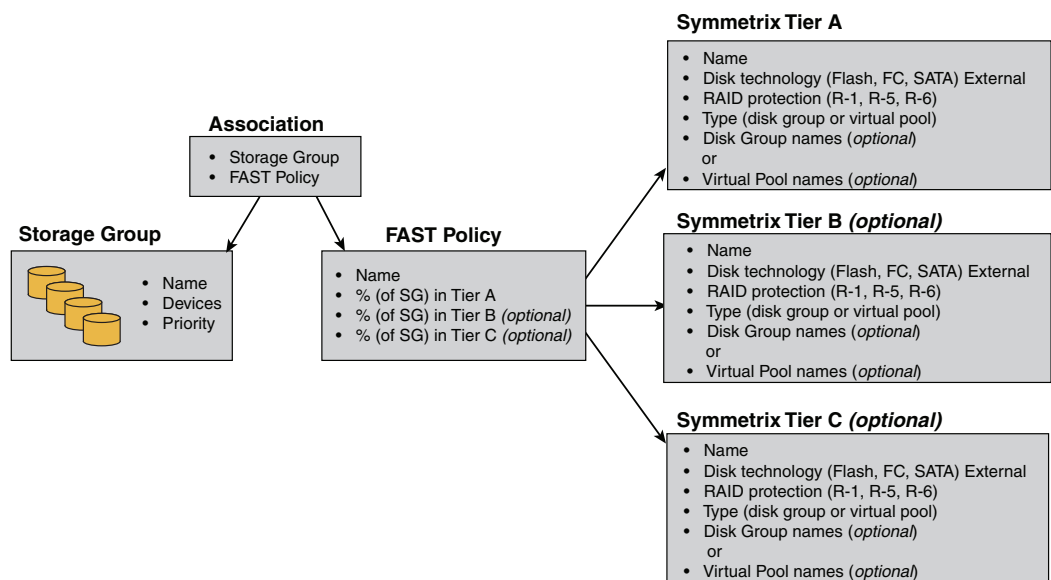
**Table 15** FAST version differences

FAST	FAST VP
Requires Solutions Enabler 7.1 and higher	Requires Solutions Enabler 7.2 and higher
Requires Enginuity 5874	Requires Enginuity 5875 and higher
Supports standard devices	Supports thin devices
Supports FBA and CKD device emulations	Supports FBA device emulation Enginuity 5876 supports thin CKD 3390 and thin IBM i 512-byte D910 devices
Disk group provisioning (DP) tiers: contain disk groups	Virtual pool (VP) tiers: contain thin pools
DP modes: Auto Approve and User Approve	VP modes: Auto Approve or None
User visible data movement plans and history	No plans or history generated
Federated Tiered Storage (eDisks) not supported	Supports Federated Tiered Storage (eDisks) with Solutions Enabler V7.4 and higher.
Thin device/thin pool compression not supported	Supports compression for thin devices and thin (VP) pools with Solutions Enabler V7.5 and higher.
Three tiers per policy supported.	Four tiers per policy supported with Enginuity V7.5 and higher. One of the four tiers must be the FTS tier.

To configure the Symmetrix array for FAST DP, the following actions are taken:

- ◆ Symmetrix tiers are defined. A Symmetrix *tier* is a specification of a type of storage (EFD, FC, SATA, eDisk), and a specification of a set of resources (disk groups/virtual pools) from which the storage will be selected. The configuration rules and restrictions for Symmetrix tiers are explained in [“Managing Symmetrix tiers” on page 178](#).
- ◆ FAST policies are defined. A FAST *policy* groups from 1 to 3 Symmetrix tiers and assigns an upper limit for each tier. The upper limit specifies how much of an associated storage group can reside on the tier. The configuration rules and restrictions for policies are explained in [“Managing policies” on page 188](#).
- ◆ Storage groups are defined. A *storage group* is a set of devices. A storage group is associated with a FAST policy and assigned a priority. A storage group can only be associated with one policy, however, one policy may be associated with many storage groups. The configuration rules and restrictions for storage groups (as related to FAST) are explained in [“Managing storage groups” on page 193](#).

The associations between a storage group, FAST policy, and defined Symmetrix tiers are shown in [Figure 14 on page 171](#).



**Figure 14** FAST component associations

- ◆ Time windows define when the FAST controller should collect performance information or execute data movement. Time windows are explained in [“Setting time windows” on page 296](#).
- ◆ Control parameters define the numbers of and types of devices, the mode of operation, the thresholds for data movement, and the analysis time period. The control parameters are explained in [“Managing the FAST controller” on page 200](#).

In addition, other tools are available to monitor your FAST configuration, including the following:

- ◆ Reports: The *compliance report* shows if the current placement of the devices in the storage group complies with the policy definition. The *technology demand report* shows the demand on each tier and technology, as per the current FAST configuration. These reports are explained in [“FAST reports” on page 214](#).
- ◆ Many `list` and `show` commands are available to examine the details of the FAST configuration. These displays are shown in [“Displaying FAST information” on page 211](#) and [“FAST Output Examples” on page 439](#).

The SYMCLI commands for FAST are supported only on Symmetrix arrays that are attached locally to the host.

---

**Note:** To use FAST DP or FAST VP, the license SYMM\_VMAX\_FAST\_TIERING must be enabled. The *EMC Solutions Enabler Installation Guide* provides all the licensing information.

---

## FAST overview

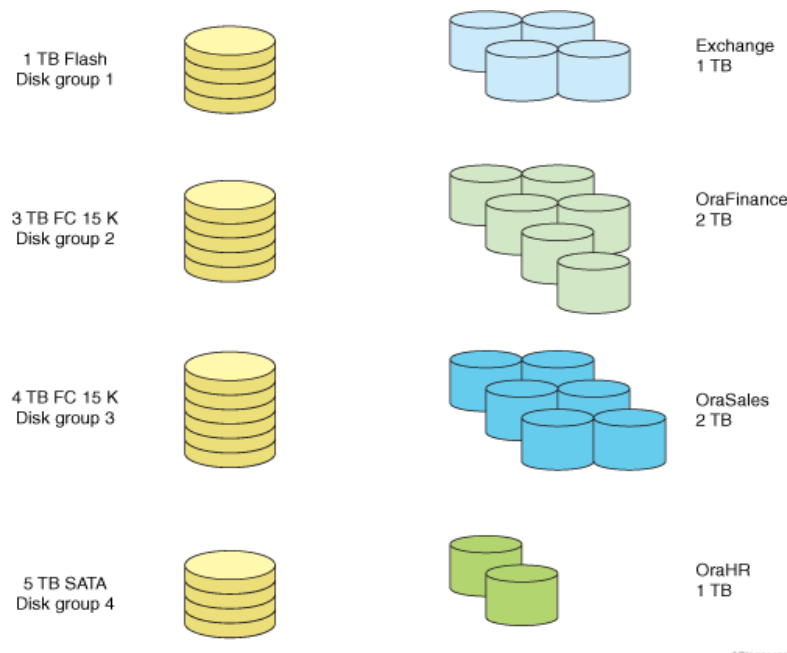
---

**Note:** FAST (disk group provisioning, or DP) supports standard devices and requires Solutions Enabler V7.1 and higher. FAST VP (Virtual Pools) supports thin devices and requires Solutions Enabler V7.2 and higher.

---

The primary goal of FAST is to share the Symmetrix disks (of varying technologies and drive speeds) among the applications so that the response time improves for the most critical applications, while the non critical applications maintain an average response time. The response time is dependent on the FAST policies that are created.

The following example uses a Symmetrix array with the simple configuration shown in [Figure 15](#).

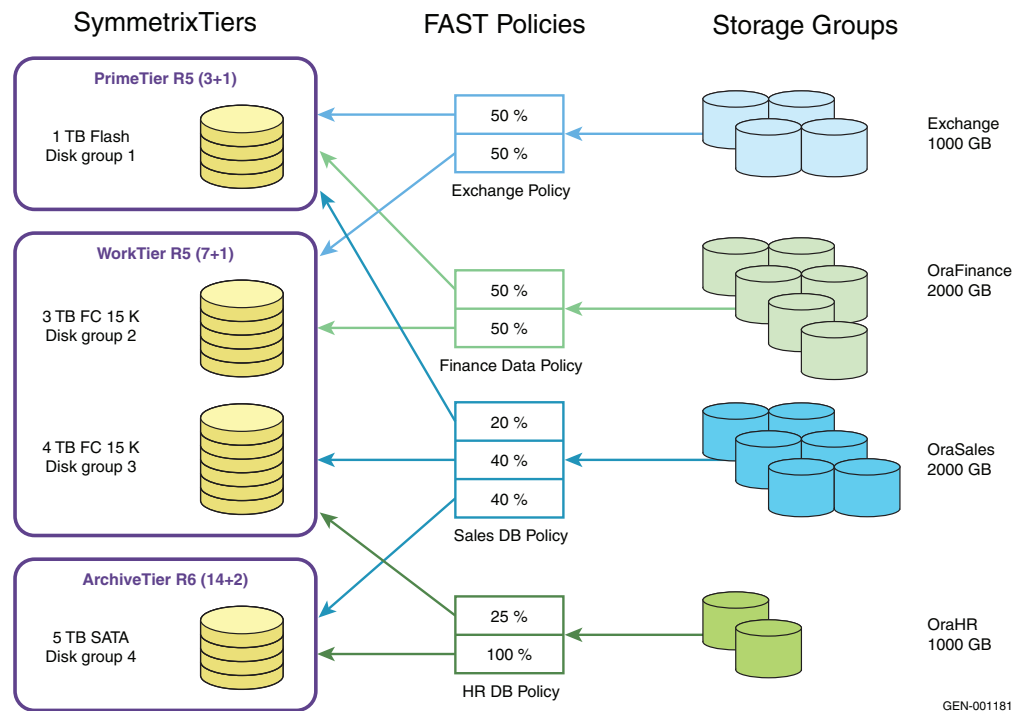


**Figure 15** Sample Symmetrix array - before FAST DP configuration

In this Symmetrix array, there are four disk groups and four applications using their storage. Flash disks are in disk group 1, Fibre Channel 15 K disks are divided into two disk groups: 2 and 3. The remaining disks are SATA, in disk group 4. All four applications use a set of these devices.

In this example, the user wants to share flash drives between three of the most important applications, Exchange, Oracle Finance, and Oracle Sales. The user also wants to make sure that the Oracle HR database does not use any of the flash space.

Figure 16 shows the completed FAST configuration. The text after the figure explains each step taken during the configuration.



**Figure 16** FAST DP configuration example

To build the configuration shown in Figure 16, the following actions are taken:

#### Tiers

The following Symmetrix tiers are configured:

- PrimeTier — With RAID 5 (3 +1) protection, Flash technology, and disk group 1.
- WorkTier — With RAID 5 (7+1) protection, Fibre Channel technology, and disk groups 2 and 3.
- ArchiveTier — With RAID 6 (14+2) protection, SATA technology, and disk group 4.

#### Policies

The following FAST policies are set:

- ExchangePolicy — PrimeTier 50%, WorkTier 50%
- FinanceDataPolicy — PrimeTier 50%, WorkTier 50%
- SalesDBPolicy — PrimeTier 20%, WorkTier 40%, ArchiveTier 40%
- HRDBPolicy — WorkTier 25%, ArchiveTier 100%

#### Storage groups

The following storage groups, containing the devices used by their corresponding applications, are created:

- Exchange

- OraFinance
- OraSales
- OraHR

### *Associations*

After the storage group is created, it can be associated with the respective FAST policy, as follows:

- Exchange — ExchangePolicy
- OraFinance — FinanceDataPolicy
- OraSales — SalesDBPolicy
- OraHR — HRDBPolicy

The association tells the FAST controller that it should optimize the performance of the storage group and also specifies the upper limits of the resources it should use to optimize performance of the devices.

For example, in the association of OraSales — SalesDBPolicy:

The FAST controller can place up to 20% of devices in the OraSales storage group on PrimeTier, up to 40% of the devices on WorkTier, and up to 40% of devices on ArchiveTier.

The capacity in a policy is specified as a percentage of the storage group, and the percentages in the policy must add up to at least 100%, but can be more than 100%. If the percentages add up to more than 100% (example `HRDBPolicy`), that means the FAST controller has some flexibility, or head room in one or more tiers, and can judiciously place the devices in the tiers deemed best suited. The percentages are only a way to limit the use of an expensive tier by the FAST controller. If there are many hot devices in the storage group, the percentage of the Flash tier can be set so that this storage group does not use all the Flash disk space at the expense of the other storage groups.

Within a FAST policy, the limit for a tier determines how many devices from an associated storage group (as a percentage of total SG logical capacity) are allowed to reside on the tier. The demand for a storage group on a tier is the physical raw capacity that the devices will need on the tier if the storage group was to occupy its full quota of space on the tier. The total demand for a tier is the sum of demands for all storage groups associated with the tier:

- ◆ 50% of the storage group `OracleFinance`, that is 1 TB of logical space or 1 TB worth of RAID 5 (3+1) devices, can be on PrimeTier. Therefore, `OracleFinance` can occupy approximately 1.33 TB ( $1 * 4/3$ ) of raw space on this tier.
- ◆ 50% of `Exchange` can be on PrimeTier that is 0.5 TB of RAID 5 (3+1) devices or 0.66TB of raw space.
- ◆ 20% of `OracleSales` can be on flash, that is 0.4 TB of RAID 5 (3+1) or 0.53 TB of raw space.

Therefore, the total demand on the tier is approximately 2.52 TB, however the tier only has 1TB of raw space. Such configurations are legal; it only means that the storage groups are competing for the resources in this tier. “FAST reports” on page 214 explains how to verify the demands on the tier in the FAST configuration.

In such cases, the priority of the storage group decides which group gets to use the resource. The user needs to specify the priority of the storage group at time of policy association. As application priorities change, these priorities can be modified.

Setting the storage group priority is explained in [“Associating a storage group to a policy” on page 196](#).

## FAST VP overview

---

**Note:** FAST supports disk group provisioning (DP) and requires Solutions Enabler V7.1 and higher. FAST VP supports virtual pools and requires Solutions Enabler V7.2 and higher.

---

FAST VP builds incrementally upon the functionality of FAST, adding support for thin devices and sub-LUN data movement. FAST VP adds finer granularities of performance measurement and data movement. The data from a single thin device under FAST control can be spread across multiple tiers. The FAST controller is free to relocate individual sub-extents of a thin device, based on performance data gathered at the extent level.

Solutions Enabler V7.5 and higher supports virtual provisioning compression, performed manually at the device level, and virtual pool VP compression, managed by the FAST controller when the compression options are set. The FAST controller provides two options for compression, time to compress and fast compression rate. [“Setting FAST control parameters” on page 203](#) provides more details about the new options for the FAST controller. [“Virtual Provisioning \(VP\) compression” on page 111](#) explains the compression feature and restrictions.

---

**Note:** A FAST VP extent is 480 thin device extents (5760 tracks, or 360 MB). The FAST VP sub-extent is 10 thin device extents (120 tracks, or 7680 KB). A FAST VP thin device extent is 1 track group (12 tracks, or 768 KB).

---

### Virtual pools

FAST VP uses the concept of virtual pools. A thin pool contains thin devices of identical emulation and protection type, all of which reside on disks of the same technology type and speed. Thin devices have no storage allocated to them when they are created; rather storage is allocated on-demand from a "bound" thin pool. The first write to a location in a thin device results in space being allocated on a DATA device from the bound pool.

With FAST VP, the data for a thin device may reside in its bound pool and potentially in one or more other pools. The role of thin pools in FAST VP is analogous to the role of disk groups in FAST. Both comprise the back-end storage available to devices under FAST control.

With Solutions Enabler V7.5, the FAST VP controller can manage compression for the virtual pools under FAST VP management. See [“Setting FAST control parameters” on page 203](#) for more details.

Refer to [“Configuring Virtual Provisioning” on page 109](#) for more information about thin pools.

### VP tiers

FAST VP uses a virtual pool (VP) tier: a set of thin pools. A VP tier has a disk technology type and a protection type. To be a member of a VP tier, a thin pool must contain only DATA devices that reside on the tier technology type and match the tier protection type. With Solutions Enabler V7.5, four tiers are allowed in a FAST VP policy.

## FTS support

Solutions Enabler V7.4 running Enginuity 5876 introduced Federated Tiered Storage (FTS). This feature virtualizes external storage as an external disk (eDisk). Adding the eDisk to the Symmetrix array makes its capacity available to the array as an external spindle.

[Chapter 5, “Federated Tiered Storage,”](#) provides additional information about FTS.

FAST VP supports tiers of externally provisioned VP pools. Encapsulated devices are not supported. There is no support for externally provisioned or encapsulated (standard) devices with FAST DP. The support for FAST VP is explained below:

- ◆ Tiers — You can create VP tiers, which can contain externally provisioned pools. The external tiers can only contain VP pools configured with externally provisioned DATA devices. With Solutions Enabler V7.4, external tiers are treated as the slowest, performance wise, technology tiers. The new order for fast to slow tiers is EFD, FC, SATA, and external tiers.

With Solutions Enabler V7.5, you can associate a technology with an FTS tier. The technology associated with the FTS tier indicates the expected performance from the tier to the FAST VP controller. This feature enables you to place the FTS tier at the right location for the expected performance of the external tier.

- ◆ Policy — You can add external tiers to a FAST VP policy. Solutions Enabler V7.5 supports up to four tiers per FAST VP policy, when one of the four tiers is an external tier.
- ◆ Association — Once a policy with an external tier is part of the association, data from the associated storage group can move to/from the external tier without any restriction.

## FAST policies

Four VP tiers in a policy is only supported on arrays running Solutions Enabler V7.5 and higher. Earlier versions of Solutions Enabler support one to three tiers in a FAST policy for both DP tiers and VP tiers.

A FAST policy is a set of one to three DP tiers or one to four VP tiers, but not a combination of both DP and VP tiers. Policies define a limit for each tier in the policy. This limit determines how much data from a storage group associated with the policy is allowed to reside on the tier.

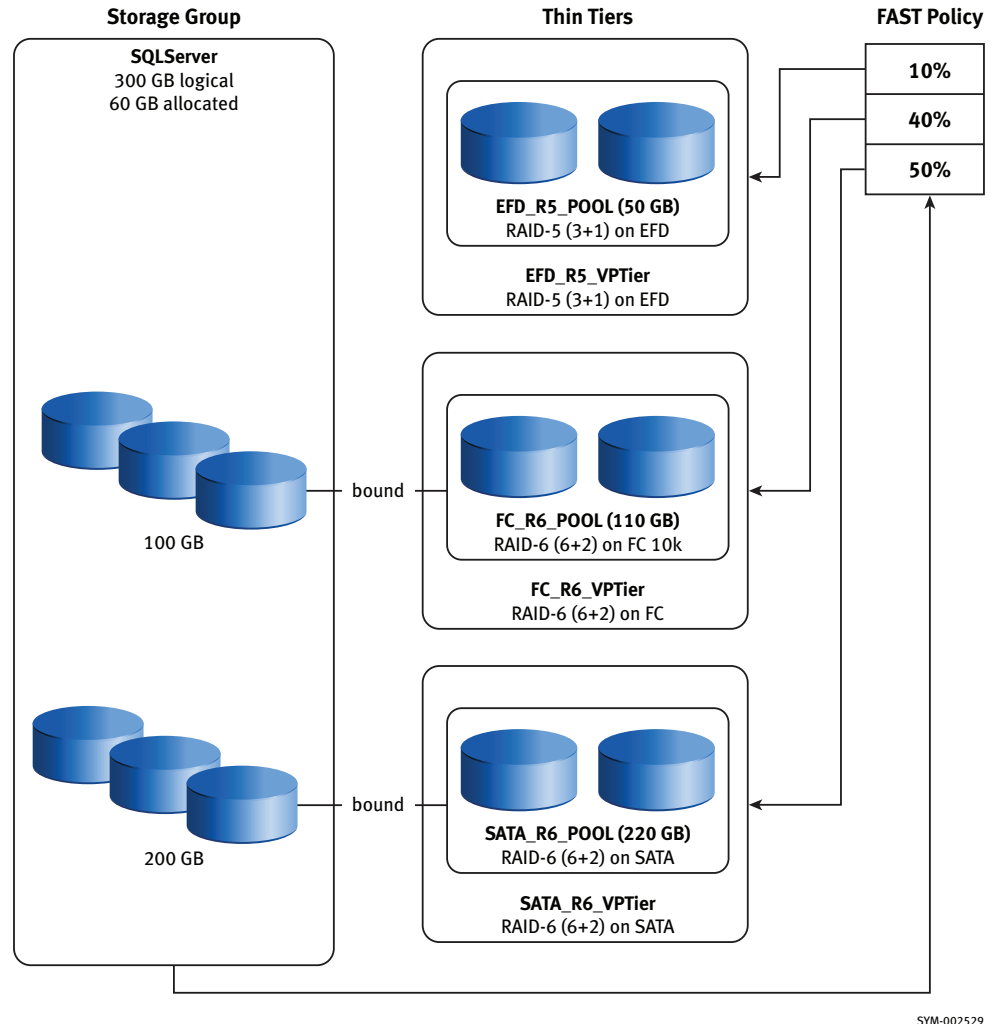
Storage groups are sets of devices. Storage groups define the devices used by specific applications. Storage groups are associated with FAST policies, and all of the devices in the storage group come under FAST control. The FAST controller can move these devices (or data from the devices) between tiers in the associated policy.

A storage group associated with a FAST policy may contain standard devices and thin devices, but the FAST controller will only act on the devices that match the type of tier contained in the associated policy. For example, if the policy contains VP tiers, then the FAST controller will only act on the thin pool devices in the associated storage group.



## FAST VP example

Figure 17 shows a simple FAST configuration in a virtual provisioned environment. The storage group `SQLServer` is associated with the FAST Policy `DBThinPolicy`. `SQLServer` contains 300 GB of thin devices.



**Figure 17** FAST VP configuration example

The following Symmetrix thin tiers were configured:

### Tiers

- `EFD_R5_VPTier` — With RAID 5 (3+1) protection, Flash technology, and thin pool `EFD_R5_POOL`. This tier contains a single pool, `EFD_R5_POOL`, which contains 50 GB of RAID-5 (3+1) DATA devices.
- `FC_R6_VPTier` — With RAID 6 (6+2) protection, Fibre Channel technology, and thin pool `FC_R6_POOL`.
- `SATA_R6_VPTier` — With RAID 6 (6+2) protection, SATA technology, and thin pool `SATA_R6_POOL`.

### Storage group

The `SQLServer` storage group contains 300 GB logical capacity, but only 60 GB of the 300 GB total logical size of `SQLServer` is actually allocated.

*Policy* The DB\_VP\_Policy is set as follows:

- EFD\_R5\_VPTier — 10%
- FC\_R6\_VPTier — 40%
- SATA\_R6\_VPTier — 50%

This means up to 10% of the allocated capacity of thin devices in `SQLServer` can reside in `EFD_R5_VPTier`, and up to 40% and 50% can reside in `FC_R6_VPTier` and `SATA_R6_VPTier` respectively.

*Thin pools*

- ◆ `EFD_R5_POOL` contains 50 GB of RAID-5(3+1) DATA devices and belongs to `EFD_R5_VPTier`.
- ◆ `FC_R6_POOL` contains 110 GB of RAID-6(6+2) DATA devices. 100 GB of thin devices from `SQLServer` are bound to `FC_R6_POOL`, meaning the initial allocations for those devices will be made in `FC_R6_POOL`. The `FC_R6_POOL` belongs to `FC_R6_VPTier`.
- ◆ `SATA_R6_POOL` contains 220 GB of RAID-6(6+2) DATA devices. 200 GB of thin devices from `SQLServer` are bound to `SATA_R6_POOL`. The `SATA_R6_POOL` belongs to the `SATA_R6_VPTier`.

Assuming the initial allocations all occurred in `FC_R6_POOL`, FAST could promote up to 6 GB of this allocated data to `EFD_R5_VPTier` and demote up to 30 GB to `SATA_R6_VPTier`, leaving up to 24 GB on `FC_R6_VPTier`.

## Managing Symmetrix tiers

A Symmetrix *tier* is the specification of the type of disk technology and speed (EFD, FC, SATA, or eDisk), the RAID protection type, and the specification of a set of resources (disk groups/virtual pools) from which the storage will be selected. Tiers can be created in the array for managing data performance with FAST, or as another way to group and report on the disk capacity usage.

### Disk group tiers and virtual pool tiers

A disk group provisioning (DP) tier is a set of disk groups with the same technology type. A DP tier has a target protection type. A virtual pool tier (VP) is a set of thin pools. A VP tier has a disk technology type and a RAID protection type. Members of a VP tier must be in thin pools with only DATA devices that reside on the tier technology type and match the tier protection type.

The `symtier` command provides the following capabilities for creating and managing tiers:

“Tier restrictions”

“Creating disk group tiers” on page 180

“Creating virtual pool tiers” on page 181

“Renaming tiers” on page 183

“Deleting tiers” on page 183

“Adding disk groups to a tier” on page 182

“Removing disk groups from a tier” on page 182

[“Adding thin pools to a tier” on page 183](#)

[“Removing thin pools from a tier” on page 183](#)

[“Listing and showing tier information” on page 184](#)

## Tier restrictions

The following restrictions apply to tiers:

- ◆ Symmetrix tier names cannot exceed 32 characters. Only alphanumeric characters, hyphens (-), and underscores (\_) are allowed, however, the name cannot start with a hyphen or an underscore. Each tier name must be unique per Symmetrix array (across both disk group and virtual pool tiers), ignoring differences in case.
- ◆ The maximum number of tiers that can be defined on a Symmetrix array is 256, counting both DP and VP tiers. If creating a new tier exceeds this limit, then an existing tier must be deleted before creating the new tier.
- ◆ Disk groups can only be specified when the tier include type is static. When specifying the disk group in the command, the disk group technology type should match the tier technology. Both the disk group name and the disk group ID cannot be specified in the same command.
- ◆ The protection types 3+1, and 7+1 are only valid when used with `-tgt_raid5`, and protection type 6 + 2 and 14+2 are only valid when used with `-tgt_raid6`.
- ◆ An error returns on the `symtier create` action, if there are no matching disk groups to the tier technology.
- ◆ An error returns on the `symtier create` action, if a tier already exists with the exact same tier definition. Two tier definitions match when they have the same RAID type and disk groups.
- ◆ A DP tier cannot be created if it will partially overlap with an existing tier. Two tiers partially overlap when they share only a subset of disk groups. For example, TierA partially overlaps with TierB when TierA contains disk groups 1 & 2 and TierB contains only disk group 2. (Creating TierA will fail.)
- ◆ A mix of static and dynamic DP tier definitions is not allowed for a single technology type. However, you can have both static VP tiers and dynamic DP tiers of the same technology type. Likewise, you can have static VP and DP tiers of the same technology type.
- ◆ Dynamic VP tiers are not supported.

## Virtual pool restrictions

The following restrictions apply to virtual pools in VP tiers:

- ◆ A pool cannot be included in a VP tier if the pool is empty, disabled, or not a thin pool.
- ◆ The thin pool and thin tier technology type must match, meaning all DATA devices in the pool must reside on disks of the VP tier technology type.
- ◆ All DATA devices in the pool must reside on disks of the same storage class (disk technology type and speed), regardless of the `SYMAPI_POOL_ALLOW_MIX_TYPE` option.
- ◆ The protection type of the thin pool and VP tier must match.

- ◆ The emulation type of all thin pools included in a given VP tier must all be the same. FBA, CKD 3390, and IBM i 512-byte D910 device emulations are supported for DATA devices.
- ◆ A VP tier can contain a maximum of 4 thin pools.
- ◆ A thin pool cannot be included in more than one VP tier.

## Creating disk group tiers

When creating a Symmetrix tier for disk group provisioning, you must specify a tier name, the protection type (RAID 1, RAID 5, RAID 6), the technology (EDF, FC, SATA), and how the disk groups will be added to the tier. There are two ways disk groups can be added to a tier:

- ◆ **Dynamic** — A `dynamic` tier automatically includes all disk groups currently in the array that match the tier technology; this type of tier will expand to accommodate any newly added disk groups.
- ◆ **Static** — A `static` tier must be manually populated with disk groups. You can use disk group name(s) or disk group ID(s). Both disk group name and ID cannot be specified using the same command. If the *DiskGroupID* is not specified, an empty tier will be created.

The `-interval` and `-count` options can be used with the `symtier create` command to retry the command in case the SYMCLI fails to get an exclusive lock on the SYMAPI database.

When the FAST controller moves a device to a tier, it changes the device's protection type to match the tier's protection type.

Use the following form to create a tier:

```
symtier -sid SymmID [-i Interval] [-c Count]
  create -name TierName
    <-tgt_raid1 |
      -tgt_raid5 -tgt_prot <3+1 | 7+1> |
      -tgt_raid6 -tgt_prot <6+2 | 14+2>>
  -technology <EFD | FC | SATA>
    -inc_type dynamic | [static
      [-dsk_grp <DiskGroupID [, DiskGroupID...] |
        name:DiskGroupName [, DiskGroupName...]>]]
```

**Example** To create a `static` tier on Symmetrix 207 called `Primary` with RAID 1 EFD technology using disk group 1, enter:

```
symtier -sid 207 create -name Primary -inc_type static -tgt_raid1 -technology EFD -dsk_grp 1
```

The new tier, `Primary`, now contains all the disks in disk group 1, resides on flash drives, and has RAID 1 protection. All disks in disk group 1 are now a part of this tier. When a storage group is associated with this tier through a FAST policy, the FAST controller may decide to move one or more devices to this tier. After the devices are moved to this tier, the devices will be in disk group 1 and their protection type will be RAID 1.

## Creating virtual pool tiers

When creating a virtual pool (VP) tier, you must specify a tier name, the protection type (RAID 1, RAID 5, RAID 6), the technology (EDF, FC, SATA, external), and thin pools names. To be a member of a VP tier, a thin pool must contain only DATA devices that reside on the tier technology type and match the tier protection type.

Thin tiers can only be static, therefore the `-inc_type` option is not included in the syntax. To create a thin tier, use the following form:

```
symtier -sid SymmID [-i Interval] [-c Count]
create -name TierName
    <-tgt_unprotected | -tgt_raid1 |
    -tgt_raid5 -tgt_prot <3+1 | 7+1> |
    -tgt_raid6 -tgt_prot <6+2 | 14+2>>
    <-technology <EFD | FC | SATA> |
    [-external]
    -vp
    [-pool <PoolName [, PoolName...]>]
```

Where:

`-technology` — Specifies the technology type for the tier.

`-external` — Indicates the tier is external to the Symmetrix array.

---

**Note:** With Solutions Enabler V7.5, you can set the technology type on an FTS tier using both the `-external` and `-technology` options. SATA will be the default technology if a technology is not specified.

---

*Example* To create a VP tier named `EFD_R5_VPTier` that has RAID 5 (3+1) protection and SATA disks on Symmetrix 234, enter:

```
symtier -sid 234 create -name EFD_R5_VPTier -tgt_raid5 -tgt_prot 3+1
-technology SATA -vp
```

## FTS tier restrictions

The following restrictions apply to creating external virtual tiers with Federated Tiered Storage (FTS):

- ◆ External tiers cannot be created with Enginuity versions lower than 5876.
- ◆ The `-tgt_unprotected` option is only valid for external tiers. In addition, it must be specified as the protection type for any external tier.
- ◆ *Solutions Enabler V7.4 only:* The `-external` option cannot be used with the `-technology` option; the technology is either EFD, FC, or SATA. The `-external` option indicates an eDisk configured using Federated Tiered Storage (FTS). This restriction has been lifted with Solutions Enabler V7.5.
- ◆ External tiers (FTS) are only supported for FAST VP.
- ◆ If there are IBM i 512-byte D910 devices in the storage group, an external tier is not allowed as part of the associated policy.
- ◆ The optionally supplied VP pools must contain external-provisioned DATA devices when creating external tiers.

- ◆ The optionally supplied VP pools must not contain external-provisioned DATA devices when creating non-external tiers.

## Adding disk groups to a tier

A disk group can only be added to a DP tier if the tier type is `static`. Use the following form to add disk groups to a tier:

```
symtier -sid SymmID [-i Interval] [-c Count]

    add -tier_name TierName [-propagate]
        -dsk_grp <DiskGroupID[,DiskGroupID...]> |
            <name:DiskGroupName[,DiskGroupName...]>
```

Tiers in a Symmetrix array cannot partially overlap. (See next *example*.) The `-propagate` option addresses the partial overlap restriction. When this option is used with the `add -dsk_grp` command, the new disk group will also be added to all tiers that overlap with the tier being modified. This option only works when adding an unused disk group to a tier. That is, a disk group that does not participate in any tier.

### Example

For example, if there are two tiers, TierA with disk groups 1 and 2, and TierB with disk group 1 and 2, the following command can be issued:

```
symtier -sid 207 add -dsk_grp 3 -tier_name TierA -propagate
```

This command adds disk group 3 to TierA, TierB, and any other overlapping tiers.

---

**Note:** If TierA had only disk group 1, and TierB had only disk group 2, Solutions Enabler does not provide a way to make TierA contain disk group 1 and 2 and TierB contain disk group 1 and 2. The only way to do this is to delete the tiers and start again.

---

## Removing disk groups from a tier

A disk group can be removed from a DP tier if the tier type is `static`. Use the following form to remove a disk group from a tier:

```
symtier -sid SymmID [-i Interval] [-c Count]
    remove -tier_name TierName [-propagate]
        -dsk_grp <DiskGroupID[,DiskGroupID...]> |
            <name:DiskGroupName[,DiskGroupName...]>
```

### Example

To remove disk group 2 from PrimeTier, enter:

```
symtier -sid 207 remove -dsk_grp 2 -tier_name PrimeTier
```

As with adding a disk group to a tier, you can use the `-propagate` option to remove the disk group from all tiers that overlap. The last disk group can not be removed from a tier that is part of a FAST policy.

## Modifying a VP tier

With Solutions Enabler V7.5, the `modify` option has been added to the `symtier` command, as follows:

```
symtier -sid SymmID [-i Interval] [-c Count]
      modify -tier_name TierName
      -technology <EFD | FC | SATA>
```

Use the `-technology` option to specify the tier type when modifying an external tier.

## Adding thin pools to a tier

Use the following form to add a thin pool to a VP tier:

```
symtier -sid SymmID [-i Interval] [-c Count]
      add -tier_name TierName
          -pool <PoolName[, PoolName...]>
```

---

**Note:** An error returns if you try to add a thin pool that is already included the tier.

---

*Example* To add thin pool `EFD_R5_POOL` to the `EFD_R5_VPTier` on Symmetrix 234, enter:

```
symtier -sid 234 add -pool EFD_R5_POOL -tier_name EFD_R5_VPTier
```

## Removing thin pools from a tier

Use the following form to remove a thin pool from a VP tier:

```
symtier -sid SymmID [-i Interval] [-c Count]
      remove -tier_name TierName
          -pool <PoolName[, PoolName...]>
```

You cannot remove all thin pools from a VP tier if the tier is part of a FAST policy. In addition, you cannot remove a pool from a tier if the tier is in an associated policy, and a device in the storage group is bound to the pool.

## Renaming tiers

A tier name can be changed using the `symtier rename` command, as follows:

```
symtier -sid SymmID [-i Interval] [-c Count]
      rename -tier_name TierName -name NewTierName
```

*Example* To rename the tier `Primary` to `PrimeTier`, enter:

```
symtier -sid 207 rename -tier_name Primary -name PrimeTier
```

Any new name must adhere to all [“Tier restrictions” on page 179](#).

## Deleting tiers

Before deleting a tier, make sure it is not associated with any FAST policy. Use the following form to delete a tier:

```
symtier -sid SymmID [-i Interval] [-c Count]
      delete -tier_name TierName [-force]
```

The `-force` option is required to delete a non-empty static tier.

**Example** To delete a tier named DBTier, enter:

```
symtier -sid 207 delete -tier_name DBTier
```

The DBTier is deleted from Symmetrix 207.

## Listing and showing tier information

The `symtier list` command gives a summary of all the disk group or virtual pool tiers in the array. The `symtier show` command includes more details about each tier.

```
symtier [-sid SymmID] [-v] [-offline]
  list [-dp | -vp [-ckd] [-fba]]
      [-technology <EFD | FC | SATA> | -external]
```

```
symtier show -sid SymmID [-offline]
```

**Note:** Prior to using the `-offline` option, the SYMAPI database must be populated with the tier information by using the command `symcfg sync -fast`.

When listing the tier information (without the `-v` option), you can specify `-dp` (disk group provisioning) or `-vp` (virtual pool provisioning). When neither option is specified, all tiers display without their capacity information.

The `symtier list` display contains two columns, one for the emulation type of FAST VP tiers and one for disk location (internal or external). FAST tiers and empty FAST VP tiers will display N/A.

**Example** To list all the tiers on Symmetrix 432, enter:

```
symtier list -sid 432
```

```
Symmetrix ID           : 000194900432
```

```
-----
Tier Name              L      Target      Flgs
                      o      Protection  ID
Type c Tech
-----
HR_DP                  DP   I SATA RAID-1    N/A   S-
HR_TEST_TIER           VP   I SATA RAID-1    FBA   S-
test_366859            VP   I SATA RAID-1    3390  S-
test_external          VP   X N/A Unprotected  FBA   SX
```

Legend:

Tier (Type) : DP = Disk Group Provisioning, VP = Virtual Pools

Disk (Loc)ation : I = Internal, X = External

Flgs:

(I)nc Type : S = Static, D = Dynamic

When the `-dp` option is specified, only the disk group tiers display, including their capacity information. When the `-vp` option is specified, only the virtual pool tiers display, including their capacity information.

**Example** To list all the thin tiers on Symmetrix 432, enter:

```
symtier list -vp -sid 432
```

```
Symmetrix ID           : 000194900432
```



Tier Name	L o c	Tech	Target Protection	Emul	Flgs ID	Logical Capacities (GB)		
						Enabled	Free	Used
HR_TEST_TIER	I	SATA	RAID-1	FBA	S-	4	4	0
test_366859	I	SATA	RAID-1	3390	S-	0	0	0
test_external	X	N/A	Unprotected	FBA	SX	4	4	0

Legend:

Disk (Loc)ation : I = Internal, X = External

Flgs:

(I)nc Type : S = Static, D = Dynamic

The columns in the output are defined as follows:

Tier Name — The name of the tier.

Loc(ation) — Internal or external tier.

Tech — Technology of the tier.

Target Protection — Tier protection type.

Emul — Device emulation type.

Flgs ID — Specifies whether the tier is static or dynamic and whether the tier is enabled for dynamic discovery.

Enabled Capacity (Logical) — Total pool enabled capacity for all thin pools in the tier.

Free Capacity (Logical) — Total enabled capacity minus the used capacity, (minimum 0).

Used Capacity (Logical) — Total pool allocated capacity for all thin pools in the tier. Allocated capacity on all DATA devices will be counted, including DATA devices that are not enabled; therefore Used may be greater than Enabled.

The next example lists the disk group tiers on Symmetrix 234:

**symtier -sid 234 list -dp**

Symmetrix ID : 000194900234

Tier Name	Tech	Target Protection	I n c	Logical Max Cfg (GB)	Raw Unconfig (GB)	Logical Config (GB)
ArchiveTier	SATA	RAID-5 (7+1)	D	78464	88113	1365
PrimeTier	EFD	RAID-1	S	645	290	500
WorkTier	FC	RAID-5 (3+1)	D	2753	1030	1980

Legend:

Inc Type : S = Static, D = Dynamic

The columns unique to the disk group tier display follow:

Logical Max Cfg (GB) — An estimate of the potential maximum logical configured capacity for the tier, if all the unconfigured space in the tier disk groups was used to create one large device of the tier protection type. The tier raw unconfigured capacity is multiplied by a factor based on tier protection type to estimate how many more logical GB worth of devices could reside on the tier (multiplied by 7/8 for a RAID-5

(7+1) tier). The resulting estimate is added to the current tier logical configured capacity to get the maximum configurable capacity. If a disk group in the tier does not have enough usable disks to support devices of the tier protection type (disk count of 7 for a RAID-5 (7+1) tier), that disk group does not contribute any capacity towards the tier maximum configurable capacity.

This value is only an estimate, and it is not guaranteed that the full maximum configurable capacity can be reached for the tier, because the estimate does not account for:

- Affinity groups.
- The physical layout of hypers on the disks.
- If standard tiers overlap, the unconfigured space from disk groups included in both tiers will be attributed to each tier, therefore the same unconfigured space may be counted multiple times.

**Raw Unconfig (GB)** — The unconfigured space in the tier disk groups. If disk group tiers overlap, the unconfigured space from disk groups included in both tiers will be attributed to each tier, therefore the same free space may be counted multiple times. If a disk group does not have enough usable disks to support devices of the tier protection type (disk count of 7 for a RAID-5 (7+1) tier), raw free capacity for that disk group will be reported as 0.

**Logical Config (GB)** — The sum of the logical capacity of all devices that match the tier protection type and reside on the tier disk groups.

## Verbose listing

Use the `-v` option to expand the output to include more details, as shown in the following command and output example:

```
symtier list -sid 432 -v
```

```
Symmetrix ID           : 000194900432
```

```
Tier Name              : HR_DP
Tier Type              : DP
Technology             : SATA
Target Protection      : RAID-1
Emulation              : N/A
Include Type          : Static
```

```
Disk Groups(1)
{
  -----
  Dsk Dsk      Speed  Disk   Logical   Raw   Logical
  Grp Grp  Name   (RPM) Count   Max Cfg Unconfig Config
  ---
  002 HELLO      7200    16     7176    11852    1250
  -----
  Total              16     7176    11852    1250
}
```

```
Tier Name              : HR_TEST_TIER
Tier Type              : VP
Technology             : SATA
Target Protection      : RAID-1
Emulation              : FBA
Include Type          : Static
```

```
Thin Pools(1)
{
-----
              Logical Capacities (GB)
Pool Name    Dev  ----- Full
              Emul  Enabled   Free   Used  (%)
-----
HR_TEST      FBA      4       4       0     0
Total
              4       4       0
}
Tier Name      : test_366859
Tier Type      : VP
Technology     : SATA
Target Protection : RAID-1
Emulation      : 3390
Include Type   : Static
```

```
Thin Pools(1)
{
-----
              Logical Capacities (GB)
Pool Name    Dev  ----- Full
              Emul  Enabled   Free   Used  (%)
-----
test_ckd     3390      4       2       2    50
Total
              4       2       2
}
```

Legend:

Tier Type : DP = Disk Group Provisioning, VP = Virtual Pools

## Show tiers

In the show display, a new `Tier Type` field displays. In addition, for thin tiers, a new `Thin Pools` table displays, as shown in the following example command and output:

```
symtier -sid 432 show -tier_name AS400_TIER
```

```
Symmetrix ID      : 000194900432

Tier Name         : AS400_TIER
Tier Type         : VP
Technology        : SATA
Target Protection : RAID-1
Emulation         : FBA
Include Type      : Static

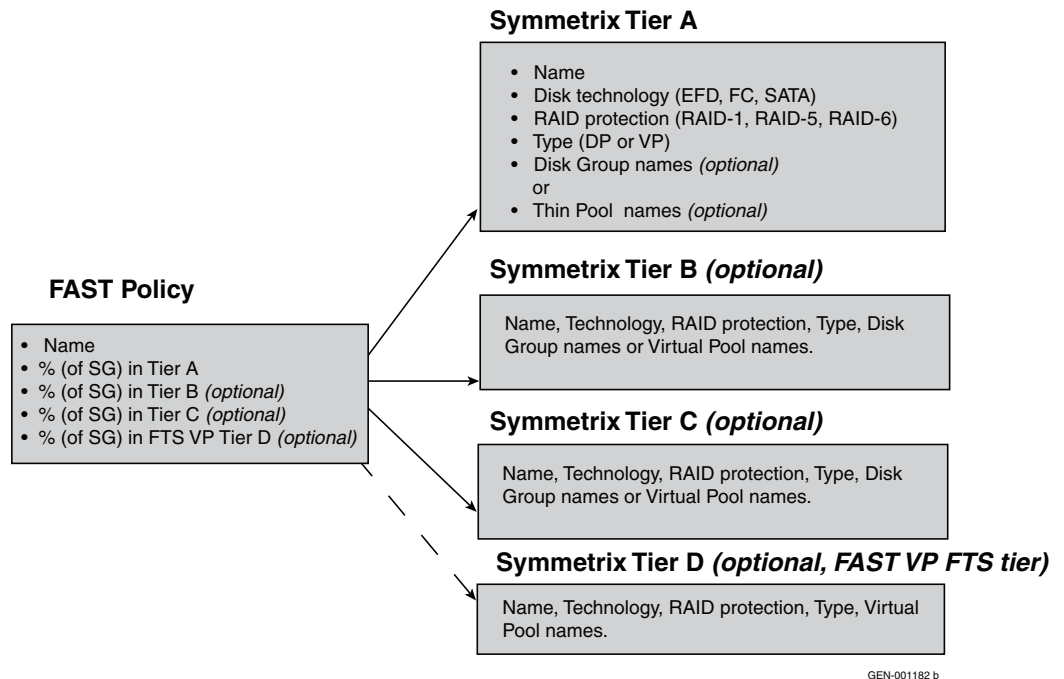
Thin Pools(1)
{
-----
              Logical Capacities (GB)
Pool Name    Dev  ----- Full
              Emul  Enabled   Free   Used  (%)
-----
AS400_TP     FBA      4       2       2    50
Total
              4       2       2
}
```

Legend:

Tier Type : DP = Disk Group Provisioning, VP = Virtual Pools

## Managing policies

A FAST *policy* is a grouping of 1 to 3 tiers (or 1 to 4 tiers for VP FTS tiers) and an assigned upper limit of how much each associated storage group (application) or VP pool can use from the tier. When creating a FAST policy, the most important component that must be set is percentage of the storage group that can reside on each tier.



**Figure 18** FAST policy overview

This section explains the following policy topics:

[“Policy restrictions”](#)

[“Creating a FAST policy” on page 189](#)

[“Adding a tier to a policy” on page 190](#)

[“Modifying the limit for a tier in a policy” on page 190](#)

[“Removing a tier from a policy” on page 191](#)

[“Renaming a policy” on page 191](#)

[“Deleting a policy” on page 191](#)

[“Listing and showing policies” on page 192](#)

### Policy restrictions

The following restrictions apply to FAST policies:

- ◆ Policy names must be unique and cannot exceed 32 characters. Policy names are case insensitive, in other words, HRPOLICY and hrpolicy are not unique names.
- ◆ Only alphanumeric characters, hyphens ( - ), and underscores ( \_ ) are allowed. The policy name cannot start with a hyphen or underscore.

- ◆ An array can have up to 256 policies.
- ◆ There can be a maximum of up to four VP tiers in a storage policy, and each tier must be unique, there can be no overlapping disk groups. For Solutions Enabler V7.4 and earlier versions, the maximum is three tiers in a storage policy.
- ◆ A policy cannot have an empty tier.
- ◆ A FAST policy may contain up to three disk group (DP) tiers or up to four virtual pool (VP) tiers, but not a combination of both DP and VP tiers. The first tier that is added to a given policy determines the type of tier that policy will contain. The fourth tier in a FAST VP policy can only be an external FTS tier.
- ◆ The emulation type of included thin pools must match across all VP tiers in a FAST policy. For example, A FAST policy cannot contain both VP tiers with FBA thin pools and VP tiers with CKD thin pools.
- ◆ If specified, the capacity limit for the tier (*MaxSgPercent*) must to be greater than 0 and less than or equal to 100. All other values will be rejected.

## Creating a FAST policy

FAST policies are created using the `symfast` command, as follows:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
      create -name FastPolicyName [-tier_name TierName
      [-max_sg_percent MaxSgPercent]]
```

The `-emulation` option is only available for arrays running Enginuity 5876 and higher. If the emulation type is not provided, the default emulation is FBA. Mixed emulation VP types are not allowed in the same policy.

The *MaxSgPercent* specified in a FAST policy indicates the upper limit of how much of the total storage group's logical capacity can be allocated to a specific tier.

For example, the assigned capacity for Tier 1 is 40% and the storage group associated with the FAST policy has five devices, each of them 40 GB. So the total capacity is 200 GB and the space available in Tier 1 for this storage group is at most 80 GB. Consequently, the FAST controller will not place more than two devices on Tier 1.

---

**Note:** The `-max_sg_percent` option is no longer required. If it is not specified, the default value is 100.

---

The FAST controller uses this capacity percent as an upper limit and does not exceed the limit. In the above case if FAST policy had 50% capacity for Tier 1, that is 100 GB for Tier 1. The FAST controller would still only place at maximum two devices on the Tier 1 since adding one more device will exceed the upper limit of 50%.

To create a FAST policy, you must supply the Symmetrix ID, a policy name, at least one tier name, and the upper limit (percent) of the tier that the storage group can occupy.

---

**Note:** The limit for the tier in the policy, `-MaxSgPercent`, will be computed using only the capacity of thin devices in the associated storage group; standard devices in the storage group are excluded from the calculation. Additionally, only allocated tracks from the thin devices will be counted, not logical device capacity.

---

*Example* To create a policy with the name `DBPolicy`, add the tier `PrimeDBTier` to the policy, and specify a capacity of 30% of a storage group, enter:

```
symfast -fp -sid 207 create -name DBPolicy -tier_name PrimeDBTier -max_sg_percent 30
```

As per the policy, no more than 30% of a storage group associated with this policy will reside on `PrimeDBTier`.

## Adding a tier to a policy

The following restrictions apply when adding a tier to a policy:

- ◆ An empty tier cannot be added; the maximum tiers for each policy is three.
- ◆ A new tier cannot be added if it will result in a configuration where two tiers in the same policy share a common disk group.
- ◆ When adding a VP tier to a FAST policy, the policy must be empty or contain only VP tiers.
- ◆ A policy can only have the same emulation VP tiers.
- ◆ A given VP tier may be included in multiple FAST policies.

To add a tier to a policy, use the following form:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
      -fp_name FastPolicyName
      add -tier_name TierName
      -max_sg_percent MaxSgPercent
```

*Example* To add the tier `ArchiveDBTier` to the policy `DBPolicy` and specify that the tier can contain up to 10% of the storage group associated with this policy, enter:

```
symfast -fp -sid 207 add -tier_name ArchiveDBTier -max_sg_percent 10 -fp_name DBPolicy
```

For any storage group associated with the `DBPolicy`, no more than 10% of its storage should be in `ArchiveDBTier`.

## Modifying the limit for a tier in a policy

To modify the limit of the tier capacity available to a storage group, use the following form:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
      -fp_name FastPolicyName
      modify -tier_name TierName
      -max_sg_percent MaxSgPercent
```

*Example* To change the limit of `PrimeDBTier` to 70% in `DBPolicy` on Symmetrix 207, enter:

```
symfast -fp -sid 207 modify -fpname DBPolicy-tier_name PrimeDBTier -max_sg_percent 70
```

As per the policy, no more than 70% of the storage group associated with this policy will reside on `PrimeDBTier`. If the policy is associated, the sum of the limits must still be equal to or greater than 100%.

## Removing a tier from a policy

Before removing any tier from a policy, make sure the `remove` action will not result in the sum of the tier limits being less than 100%. If the tier is the only tier in the policy, and the policy is associated with a storage group, the tier cannot be removed. A VP tier cannot be removed from a policy if thin devices in the associated storage group are bound to pools in the tier.

Use the following form to remove a tier from a policy:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
        -fp_name FastPolicyName
        remove -tier_name TierName
```

*Example* To remove `ArchiveTier` from the `FinanceData` policy on Symmetrix 207, enter:

```
symfast -fp -sid 207 -fp_name FinanceData remove -tier_name ArchiveTier
```

The `ArchiveTier` is removed from the policy.

## Renaming a policy

When renaming a policy, make sure the new name does not already exist. To rename a FAST policy, use the following form:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
        -fp_name FastPolicyName
        rename -name NewFastPolicyName
```

*Example* To rename `DBPolicy` to `OraDBPolicy` on Symmetrix 207, enter:

```
symfast -fp -sid 207 rename -fp_name DBPolicy -name OraDBPolicy
```

The new name for the policy is `OraDBPolicy`.

## Deleting a policy

The following restrictions apply to deleting FAST policies:

- ◆ A policy that has tiers can only be deleted by using the `-force` flag.
- ◆ The policy must not be associated with any storage group. If the policy is associated, attempts to delete the policy return an error. To delete the policy, first disassociate the storage group from the policy, and then delete the policy.

To delete a FAST policy, use the following form:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
        -fp_name FastPolicyName
        delete -fp_name FastPolicyName [-force]
```

*Example* To delete FAST policy `HRPolicy` on Symmetrix 207, enter:

```
symfast -fp -sid 207 delete -fp_name HRPolicy
```

The FAST policy `HRPolicy` is deleted from the array.

## Listing and showing policies

To list and show FAST policies, use the following form:

```
symfast [-sid SymmID] [-offline] [-v]
list -fp [-v] [-dp | -vp [-ckd | -fba]]

symfast [-sid SymmID] [-offline]
show -fp_name FastPolicyName
```

Notice that the `list` command includes an optional filter for disk group provisioning (`-dp`) or virtual pools (`-vp`) and the emulation type (`-ckd`) and (`-fba`).

**Note:** Prior to using the `-offline` option, the SYMAPI database must be populated with the FAST information by using the command `symcfg sync -fast`.

### Examples

In the policy output, when you specify the `-dp` option, only the policies containing disk group tiers display. When you specify the `-vp` option, only the policies containing virtual pool tiers display. When neither option is specified, all policies display.

To show the FAST policies for Symmetrix 601 that contain VP tiers, enter:

```
symfast -sid 601 show -fp_name test_policy
```

```
Symmetrix ID           : 000195700601
```

```
Policy Name            : test_policy
```

```
Emulation              : FBA
```

```
Tiers(4)
```

```
{
-----
                                     L
                                Max SG O
                                Percent C Tech  Target
Tier Name                      Type   Protection
-----
test_EFD                       VP      10 I EFD  RAID-1
test_FC                        VP      30 I FC   RAID-5(3+1)
test_SATA                      VP      40 I SATA RAID-6(6+2)
test_FTS                       VP      70 X N/A  Unprotected
}
```

```
Storage Groups(1)
```

```
{
-----
Storage Group Name            Pri
-----
test_sg                       1
}
```

Legend:

```
Tier Type      : DP = Disk Group Provisioning, VP = Virtual Pools
Disk (Loc)ation : I = Internal, X = External
```



When a FAST VP policy is listed with the `-v` option, a flags (Flgs) column indicates if the tier can be compressed, as follows:

```
symfast -sid 601 list -fp -v
```

```
Symmetrix ID      : 000195700601
```

```
Policy Name       : test_fp
```

```
Emulation         : FBA
```

```
Tiers(1)
```

```
{
-----
                                L
                                Max SG O
Tier Name                      Type  Percent C Tech  Target      Flgs
-----
test_tier                      VP      100 I FC   RAID-1      .
}
```

```
Storage Groups(1)
```

```
{
-----
Storage Group Name            Pri
-----
test_fast                      2
}
```

```
Legend:
```

```
Tier Type      : DP = Disk Group Provisioning, VP = Virtual Pools
```

```
Disk (Loc)ation : I = Internal, X = External
```

```
Flgs:
```

```
(C)ompression   : X = Compression Capable, . = Not Compression Capable
```

## Managing storage groups

Storage groups are a collection of devices on the Symmetrix array that can be used for masking/mapping, Virtual LUN Technology, and FAST operations. This section explains how to create a storage group, how to add and remove devices, and explains the FAST-related storage group options.

This section explains the following storage group topics:

[“FAST storage group and device restrictions”](#)

[“Creating storage groups” on page 195](#)

[“Adding devices and storage groups” on page 195](#)

[“Associating a storage group to a policy” on page 196](#)

[“Disassociating a storage group from a policy” on page 197](#)

[“Modifying a storage group” on page 198](#)

### FAST storage group and device restrictions

The following restrictions apply to storage groups that are associated with FAST policies:

- ◆ A storage group can be associated with only one policy. The policy must have at least one tier.

- ◆ The emulation type of all devices in the storage group, including both standard and thin devices, must be the same.
- ◆ Thin device emulation types supported by FAST VP are FBA, CKD 3390, and IBM i 512-byte D910 devices.
- ◆ The Symmetrix array can have a maximum of 8192 storage group associations to policies. Further, only 1000 of these associations can be with policies containing VP tiers.
- ◆ The total of the limits in the FAST policy must add up to at least 100%.
- ◆ A device can be associated with only one policy. An attempt to associate a storage group with a device that is already associated to a policy will be blocked. This restriction applies to all devices, standard and thin, associated with any FAST policy, regardless of whether the policy contains DP or VP tiers. For example, a standard device in a storage group associated with a policy containing VP tiers could not be added to a different storage group associated with a policy containing DP tiers. To place all of the standard and thin devices under FAST control, two separate storage groups would have to be created.
- ◆ Only the metahead is allowed in the storage group; metamembers cannot be part of a storage group.
- ◆ Storage groups containing a mix of standard and thin devices can be associated with a FAST policy containing VP tiers, however, FAST will only operate on the thin devices, and only the capacity from thin devices will be used to determine compliance. Conversely, when the FAST policy contains DP tiers, FAST will only operate on the standard devices, and only the capacity from standard devices will be used to determine compliance.

All devices in the storage group are considered under FAST control, however, a choice must be made whether FAST will manage the thin devices or the standard devices in a given storage group.

- ◆ Devices that are not movable cannot be added to a storage group that is (or will be) associated with a FAST policy. FAST does not support moving the following device types:
  - AS400, ICOS, ICL
  - CKD EAV, CKD EAV phase 3, and CKD concatenated metadevices
  - Diskless
  - DRV
  - SAVE
  - SFS
  - Unprotected
  - VDEV (can be added to a storage group, but will be ignored for FAST operations)
- ◆ If the FAST policy contains VP tiers, none of the thin devices can be bound to a pool outside of the policy.
- ◆ The FAST policy can contain multiple VP tiers with the same technology type, however FAST VP only performs *compliance* movements between such tiers. There will be no *performance* movements between tiers of the same technology type within a single FAST policy.

- ◆ No checks are made for oversubscription at association time. A virtual pool in a VP tier in an associated FAST policy becomes a shared resource; its capacity is no longer used not only by the bound thin devices but also thin devices under FAST control. Therefore, even a pool that is not oversubscribed (`MAX_SUBS_PERCENT <= 100%`) may become full due to a combination of allocations for bound devices and FAST moving chunks into the pool.

## Creating storage groups

Storage groups are created using the `symmsg` command. To create a storage group, use the following form:

```
symmsg -sid SymmID [-i Interval] [-c Count]
      create SgName [-bw_max MBperSec] [-iops_max IOperSec]
```

Where:

`-bw_max` — Limits the front-end bandwidth of the devices in the storage group. The valid range for bandwidth is from 1 MB/Sec to 100,000 MBs/sec.

`-iops_max` — Limits the I/Os per second of the devices over a set of director ports. The valid range for IOPs is from 100 IOs/sec to 100,000 IOs/sec but must be specified in units of 100 IO/Sec.

The following restrictions apply to creating storage groups:

- ◆ GNS does not support storage groups. Storage groups are saved in a special area on the Symmetrix array.
- ◆ An array can contain a maximum of 8192 storage groups.
- ◆ Each storage group can contain a maximum of 4096 devices.
- ◆ Storage groups are supported only on Symmetrix arrays running Enginuity 5874 and higher.

*Example* To create storage group `sg1` on Symmetrix 207, enter:

```
symmsg -sid 207 create sg1
```

An empty storage group, `sg1`, is created on Symmetrix 207.

## Adding devices and storage groups

To add devices or storage group(s) (requires Enginuity 5876), use the following form:

```
symmsg -sg SgName -sid SymmID [-i Interval] [-c Count]
      add dev SymDevName
      add sg SgName [,SgName2, SgName3, ..., SgNameN]
```

Refer to the *EMC Solutions Enabler Symmetrix Management CLI Product Guide* for the details and restrictions for adding and removing (child) storage groups.

To add multiple devices, or devices in a range or a file, use the following form:

```
symmsg -sg SgName -sid SymmID [-i Interval]
      [-c Count] [-v]
      [-SA <# | ALL>] [-P #] [-N #]
      [-cap # [-captype <mb> | <cyl>]]
      [-devs <SymDevStart:SymDevEnd | SymDevName
        [, <SymDevStart:SymDevEnd | SymDevName>...]> |
      -file DeviceFileName [-tgt]>]
```

```
addall [pd | devs]
```

Although the `-devs` option implies you can enter a single range or a list of comma delimited devices, the option supports a free-form combination of entries. An example of `-devs` usage is:

```
symmsg -sid 207 -sg sg1 -devs 64:105,22a,505,600:605,0700
```

There are many additional options with the `symmsg` command. The complete command set description is in the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

## Associating a storage group to a policy

**Note:** With Solutions Enabler V7.5, setting the `-priority` on an association is optional. If a priority is not provided, a default priority of 2 will be automatically set.

Use the `symfast` command to associate a storage group to an existing FAST policy. This operation associates a storage group to an existing FAST policy and assigns a priority value for the association. The priority of a storage group affects which storage group is serviced first by the tier when there is contention among FAST storage groups for space in a storage tier. The values are 1, 2, or 3. Priority 1 has the highest value, priority 3 has the lowest value.

## SRDF Coordination

Previously, when R1 devices and R2 devices were both managed by FAST, there was no coordination of FAST VP actions (moves) between the local and remote Symmetrix arrays. Both sets of devices were managed individually, based on statistics collected locally on each array. Without coordination, the R1 device and the corresponding R2 device could reside in different tiers, which could impact RDF performance.

Solutions Enabler V7.4 running Enginuity V5876 provides an option to set FAST VP RDF coordination on an association. This instructs FAST to factor the R1 device statistics into the move decisions that are made on the R2 device. This attribute can be set on a storage group, even when there are no SRDF devices in the storage group.

To associate a storage group with a FAST policy, and set any associated attributes, 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>]
```

With Solutions Enabler 7.5, setting priority on an association is optional. (Previously, the `-priority` option was required.) Solutions Enabler sets the default priority at 2 if a value is not supplied.

For the SRDF coordination, both the R1 and the R2 devices must reside on an array running Enginuity 5876 or higher.

You cannot associate a parent SG with a FAST policy.

*Example* To associate storage group `OraSales` with the FAST policy `DBPolicy`, and set the storage group priority at 1, enter:

```
symfast -sid 207 associate -sg OraSales -fp_name DBPolicy -priority 1
```

This association is successful when the storage group configuration rules have been followed and there is enough capacity allocated in the tiers that belong to the FAST policy. The association will not be blocked if the tiers do not have sufficient capacity, but the storage group will be out of compliance with its policy.

After the storage group is associated with the policy, the FAST controller will actively monitor the storage group and suggest or run optimizations to improve performance.

## Reassociating a storage group to a new policy

With Solutions Enabler V74, you can reassociate a storage group to a new policy. When a storage group is reassociated, all the current attributes set on the original association automatically propagate to the new association. This feature eliminates the previous process of disassociating a storage group, then associating the group to a new policy, and entering the attributes, such as priority, on the association.

The `reassociate` option has been added to the `symfast` command, as follows:

```
symfast -sid SymmID [-i Interval] [-c Count]
        -sg_name SgName
reassociate -fp_name FastPolicyName
```

### Restrictions

The following restrictions apply to the reassociate action:

- ◆ The storage group name must be a valid storage group name.
- ◆ The storage group and policy must already exist on the Symmetrix array.
- ◆ The storage group must be in an association before performing a reassociation.
- ◆ The new policy for the storage group, must have the same emulation as the storage group. Mix emulation association will result in an error.
- ◆ The storage group cannot be associated with empty policy, the reassociated policy must contain at least one tier.
- ◆ The total of the capacity percentage for the target FAST policy must add up to at least 100%.
- ◆ If the FAST policy contains VP Tiers, all of the thin devices in the storage group must be bound to any VP pool in a tier in the policy. None of the thin devices can be bound to a pool outside of the policy.

## Disassociating a storage group from a policy

When a storage group is disassociated from a FAST policy, the FAST controller no longer tries to optimize the performance of that storage group. To disassociate a storage group from a policy, use the following form:

```
symfast -fp -sid SymmID [-i Interval] [-c Count]
        -fp_name FastPolicyName
disassociate -sg SgName
```

**Example** To disassociate storage group `OraSales` from the FAST policy `DBPolicy`, enter:

```
symfast -sid 207 disassociate -sg OraSales -fp_name DBPolicy
```

The disassociate operation occurs, even if the FAST controller is currently moving devices in the storage group.

Disassociating a storage group containing thin devices means those devices in the storage group are no longer under FAST control. Data from the thin devices remains where it is, even if it is spread across multiple pools. FAST will not move the data back to the bound pool.

## Modifying a storage group

With Engenuity 5876, you can modify a storage group priority and the SRDF coordination. The SRDF coordination is only applicable to FAST VP.

The priority of the storage group can only be changed if the storage group is associated with a FAST policy.

To change a storage group priority or the SRDF coordination attribute, use the `symfast modify` command, as follows:

```
symfast -sid SymmID [-i Interval] [-c Count]
        -fp_name FastPolicyName
modify -sg SgName [-priority PriorityValue]
        [-rdf_coordination = ENABLE | DISABLE]
```

The priority values can be 1 (highest), 2, or 3 (lowest). The SRDF coordination attribute is only applicable for FAST VP.

**Example** To decrease the priority of storage group `OraMarketing`, associated with FAST policy `DBPolicy`, from 1 to 2, enter:

```
symfast -sid 207 modify -sg OraMarketing -fp_name HRDBPolicy -priority 2
```

The storage group `OraMarketing` has a priority of 2.

## Displaying storage group associations

Use the `symfast list -association` command to display the storage groups, their priority, their associated policies, and whether the SRDF coordination attribute is set.

**Note:** Before using the `-offline` option, the SYMAPI database must be populated with the storage group, tier, and FAST data using the `symcfg sync -fast` command.

**Examples** To display the storage group associations on Symmetrix 432, enter:

```
symfast list -sid 432 -association
```

Output similar to the following displays:

```
Symmetrix ID           : 000194900432
```

Storage Group Name	Policy Name	Pri	Flgs R
HR_SG	HR_FP	3	X
HR_TEST	HR_FP	2	.

Legend:

Flgs:

(R)DF Coordination : X = Enabled, . = Disabled

The output is sorted alphabetically by storage group name.

The `show` action of `symfast` displays information about associations between FAST policies containing standard or thin tiers and storage groups.

```
symfast [-sid SymmID] [-offline]
  show -association -sg SgName [-all] [-v]
```

The show association display includes a `Tier Type` column in the `Tiers` table. A new table lists thin devices. By default, only the devices managed by FAST display (thin devices in the case of a policy containing thin tiers, and standard devices in the case of a policy containing standard and thin tiers); gatekeeper devices are filtered out.

Use the `-all` option to show all devices in the storage group, regardless of the associated policy.

The `symfast show -association` output shows FAST VP associations as Enable or Disable. Disable displays for all FAST VP associations on Symmetrix arrays running versions lower than Enginuity 5876. FAST (DP) associations will display N/A.

**Example** To show the associations for storage group `test_fast_VP` on Symmetrix 432, enter:

```
symfast -sid 432 show -association -sg test_fast_VP
```

```
Symmetrix ID           : 000195700432
```

```
Storage Group          : test_fast_VP
```

```
Thin Devices(1)
```

```
{
-----
      Flgs Dev      Total Bound      Allocated
Sym  PC  Emul      Tracks Pool Name      Tracks
-----
06D1 NX  FBA        30945 test_fast_VP      15000
06D2 N.  FBA        30945 test_fast_FC      15000

Total
Tracks      -----
      61890
GBs          4
}
```

```
Policy Name            : test_fast_VP
```

```
Priority                : 1
```

```
RDF Coordination       : Enabled
```

```
Tiers(1)
```

```
{
-----
Tier Name                                Type  Max SG      Target      Flgs
                                Percent Tech  Protection  C
-----
test_fast_VP                        VP      100 SATA  RAID-1      X
test_fast_FC                        VP      100 FC    RAID-1      .
}
```

## Legend:

Tier Type: DP = Disk Group Provisioning, VP = Virtual Pools

## Device Flags:

(P)inned : Y = Device is Pinned, N = Device is not Pinned

(C)ompression : X = Compressed Allocation, . = No Compressed Allocation

## Tier Flags:

(C)ompression : X = Compression Capable, . = Not Compression Capable

---

**Note:** The order of the associated tiers displays highest to lowest, meaning the highest performing tier is listed on top and the lowest performing tier is listed on the bottom.

---

A verbose `show association` display can be generated. In addition to the information shown by the non-verbose display, the verbose display shows the distribution of allocated tracks from thin devices in the storage group across the thin tiers in the associated FAST policy. By default, only the devices managed by FAST display, either standard or thin devices, based on the policy. Gatekeeper devices are also filtered out.

[Appendix C, “FAST Output Examples,”](#) provides more examples of output displays.

## Managing the FAST controller

The FAST controller is the intelligence of the FAST software. The FAST controller runs its algorithms continuously in the background, checking to see if any FAST policies initiate a performance improvement for the applications managed by those policies.

The FAST controller has two states—Enabled and Disabled. Disabled means that the FAST controller does not do any data movement between tiers based on the FAST policies. Enabled means that the FAST controller performs data movement between tiers based on the policies. You can differentiate between FAST and FAST VP with the enable and disable commands by adding the `-dp` or `-vp` options to the command.

The FAST controller considers EFD as the highest performing tier, FC as the second highest, and SATA the slowest performing tier. External tiers are considered slower than SATA tiers. The FAST controller tries to place the busiest volumes on EFD, as well as make sure it does not exceed the upper limit set by the FAST policy.

In FAST, the FAST controller, an entity residing on the Service Processor, was solely responsible for FAST data movements. The FAST controller gathered and analyzed device statistics then issued explicit movement instructions to the Symmetrix array.

With FAST VP, there is a division of responsibilities between the FAST controller and the Enginuity operating system. The performance information is gathered by Enginuity and sent to the FAST controller. It issues explicit movement instructions for thin devices, but only for compliance movements. The responsibility for performance movements for thin devices rests with Enginuity. The FAST controller sends to Enginuity a set of partial instructions, called a movement policy, for each storage group under FAST control. Enginuity interprets these instructions and independently executes performance movements for thin devices, based on the device statistics.

Because of these architectural changes, in the event of a loss of the service processor, performance statistics will continue to be collected. Performance data movements will also continue for a short period of time, but will eventually stop, until the service processor is recovered.

This section explains the following FAST controller topics:



- [“Enabling and disabling FAST” on page 201](#)
- [“Displaying the FAST state” on page 202](#)
- [“Managing time windows” on page 203](#)
- [“Setting FAST control parameters” on page 203](#)
- [“Displaying FAST control parameters” on page 206](#)
- [“Displaying FAST plans” on page 207](#)
- [“Approving a FAST plan” on page 209](#)
- [“Declining a FAST plan” on page 210](#)

## Enabling and disabling FAST

By default, the FAST controller is disabled when shipped. Using the `symfast` command, the FAST controller can be enabled or disabled. To enable the FAST controller, use the `symfast enable` command.

```
symfast -sid SymmID [-i Interval] [-c Count] [-noprompt]
    enable [-dp | -vp]
    disable [-dp | -vp]
```

Where:

`-dp` — Enables/disables FAST disk group provisioning.

`-vp` — Enables/disables FAST VP (virtual pool) provisioning.

If neither option is supplied, both FAST and FAST VP will be enabled/disabled.

---

**Note:** Solutions Enabler V7.4 and higher supports FAST controls for local and remote Symmetrix arrays running Enginuity 5876.

---

### Examples

To enable the FAST controller for disk group provisioning, enter:

```
symfast -sid 207 enable -dp
```

To disable the FAST VP controller, enter:

```
symfast -sid 207 disable -vp
```

---

**Note:** Depending on the state of the FAST controller, the `disable` action can take some time to complete.

---

Disabling the FAST controller without the `-dp` or `-vp` options stops all data movement of both standard and thin devices, although if there is currently a FAST standard plan executing, the current group will complete before the FAST controller is disabled. Likewise, if the FAST controller is currently executing thin data movements for compliance, those data movements will complete before the FAST controller is disabled.

If the disable command fails, for example because the FAST controller is unresponsive or DisabledWithError, it is possible that performance movement will continue for thin devices.

In this case, the only way to completely stop the thin data movement is to change the `-thin_data_move_mode` control parameter to `NONE`. Refer to [“Setting FAST control parameters” on page 203](#).

## Displaying the FAST state

To verify the state of the FAST controller, use the `list -state` command, optionally specifying the `-vp` or `-dp` options. If `-vp` or `-dp` are not specified, both options display.

Output displays similar to the following two examples:

### Example 1

```
symfast -sid 234 list -state -dp
```

```
Symmetrix ID: 000194900234
```

```
FAST State           : Enabled
Degraded Reason      : N/A
Error Message        : N/A
Current Activities    : Idle
```

### Example 2

```
symfast -sid 234 list -state -vp
```

```
Symmetrix ID: 000194900234
```

```
FAST State           : Degraded
Degraded Reason      : No DRVs
                     FAST VP is not licensed
Error Message        : N/A
Current Activities    : Idle
```

In the previous output the `FAST State` can have the following possible values:

`Enabled` — The FAST controller is enabled.

`Disabled` — The FAST controller is disabled.

`Disabling` — The FAST controller is in transition from `Enabled` to `Disabled`. Or `Disable` is in progress.

`DisabledWithError` — The FAST controller is disabled with error.

`Degraded` — The FAST controller was activated but not fully functional because of absence of DRV devices or other reasons. It is possible for multiple degraded reasons to occur simultaneously. The `Degraded Reason` field may have a reason for the degraded state or will have a value of `N/A`. The possible values for the `Degraded Reason` are:

- FAST thin tiers reached PRC (Pool Reserved Capacity) value
- FAST VP compliance movement failed
- FAST VP performance movement policy update failed
- FAST algorithm experienced an error
- Optimizer algorithm experienced an error
- Optimizer/FAST standard move time window is not present or does not extend into the future
- FAST thin move time window is not present or does not extend into the future
- Performance time window is not present or does not extend into the future
- Optimizer/FAST movement is not allowed due to internal hard script error
- Optimizer is not licensed
- FAST is not licensed

- FAST VP is not licensed
- Statistics collection is failing for standard device; No Performance movement will happen
- Statistics collection is failing for thin devices; No Performance movement will happen
- No DRVs
- Broken metadevices
- Illegal time windows
- Illegal groups and rules
- Illegal device attributes
- Illegal FAST parameters

The `Error` Message will display more information, if available, about the FAST controller when its state is `DisabledWithError`.

In the previous output the `Current Activities` can have the following possible values:

`Idle` — Fast controller is current idle.

`RunningPlan` — The FAST controller is running a plan.

`FetchingStats` — The FAST controller is fetching statistics.

`AnalyzingStats` — The FAST controller is analyzing statistics.

`PendingPlan` — The FAST controller has a plan scheduled and is waiting to start the plan.

## Managing time windows

Solutions Enabler V7.3 introduces the `symtw` command for managing time windows. You can use this command to convert any existing time windows to the new format.

Refer to [Chapter 7, “Managing Time Windows”](#) for information about managing time windows.

## Setting FAST control parameters

FAST control parameters are set using the following form:

```
symfast -sid SymmID [-i Interval] [-c Count]
set -control_parms
    [-approval_mode Auto_Approve | User_Approve] |
    [-min_perf_period PerfTime] |
    [-vp_data_move_mode <Auto | None>]
    [-workload_period WorkTime] |
    [-max_simult_devs MaxSimultDevs] |
    [-max_devs MaxDevs] |
    [-vp_reloc_rate ThinRate]
    [-swap_notvisible_devs Enable | Disable]
    [-allow_only_swap Enable | Disable ]
    [-pool_resv_cap ResvPct]
    [-vp_allocation_by_fp <ENABLE | DISABLE>]
    [-time_to_compress <NumDays | never>]
    [-fast_compression_rate <FastCompRate>]
```

Where:

`-approval_mode` — Sets the mode of the FAST controller to automatic or user approval mode using a command line. The possible values are: `Auto_Approve` and `User_Approve`. The default value is `Auto_Approve`. If the FAST controller is set to `User_Approve` mode, it will generate plans, but not perform any movements unless the plans are approved by the user. The `-approval_mode` option can set the mode of the FAST controller and Optimizer using a command line. This parameter is for standard devices only.

---

**Note:** *With Solutions Enabler V7.1.1 and higher, the `-approval_mode` option replaces the `-mode` option. The `-mode` option will continue to work, although its use will be deprecated. The `-mode` option can also be used to set the CLI compatibility mode.*

---

`-min_perf_period` — Specifies the amount of samples required initially before a recommendation is made. You should make sure that the values you specify are long enough (usually a week) for FAST to establish a good characterization of your typical workloads. The parameter exists in case you do not want to wait until the entire workload period has elapsed before the FAST controller commences its analysis and activity. The minimum value is 2 hours, the maximum value is the current value of the workload period parameters.

`-vp_data_move_mode` — The VP data movement mode for thin devices is analogous to the approval mode for standard devices. In `Auto` mode, the FAST system continuously performs data movement for thin devices within the data movement window, without user intervention. If the mode is set to `None`, the FAST controller does not perform any data movement for thin devices. There is no equivalent of `User_Approve` mode for VP data movement.

`-workload_period` — Specifies the amount of workload sampling that FAST should maintain for sample analysis. This is specified in units of time (hours). For example, you can indicate the FAST controller should maintain two weeks worth of workload samples for analysis. The minimum value is 2 hours, the maximum value is 672 (4 weeks).

`-max_simult_devs` — Determines the maximum number of devices that can be moved simultaneously. The minimum value is 2, the maximum value is 32. The default value is 8. This parameter is for standard devices only.

`-max_devs` — Determines the maximum number of devices that can be moved in a 24-hour period. The minimum value is 2, the maximum value is 200. The default value is 200. This parameter is for standard devices only.

`-vp_reloc_rate` — The FAST VP relocation rate determines the aggressiveness of the data movement for thin devices, similar to the `max_simult_devs` setting for standard devices. The lower the value of the VP relocation rate, the more aggressive FAST will be. This setting does not affect the speed of the data movement itself; that is controlled by the VLUN QoS setting. The minimum value is 1, the maximum value is 10, and the default value is 5.

`-swap_notvisible_devs` — Indicates if the FAST controller can use host invisible devices (unmasked and unmapped) to do a full swap with devices in storage groups under FAST control to improve performance of the storage group. Possible values are `ENABLE` and `DISABLE`. The default is `DISABLE`. This parameter is for standard devices only.

`-allow_only_swap` — Indicates that the FAST controller can only perform a full swap of devices and cannot move devices to unconfigured space. The default is DISABLE. This parameter is for standard devices only.

`-pool_resv_cap` — The pool reserved capacity (PRC) is a percentage of the capacity of each virtual pool that will be reserved for non-FAST activities. If the free space in a given pool (as a percentage of pool-enabled capacity) falls below the PRC, the FAST controller does not move any more chunks into that pool. To move any new chunks to the pool, the FAST controller must first move some chunks from that pool to another pool to free up space. Enforcement of the PRC is best-effort; FAST may move chunks to a virtual pool resulting in a violation of the PRC, because non-FAST activities (such as new allocations for writes to a thin device) can simultaneously consume pool free capacity. The minimum value is 1, the maximum is 80, and the default is 10.

---

**Note:** To set the PRC of an individual pool, use the `set pool` command, as explained in [“Setting pool attributes” on page 121](#). If the PRC is not set at the pool-level, the value set with `symfast` is the default.

---

`-vp_allocation_by_fp` — When set to ENABLE, for all thin devices managed by FAST, Enginuity will choose a pool from the policy when making an allocation for the thin device. When set to DISABLE, the allocation will be from the bound pool. The default value for this option is DISABLE.

---

**Note:** The `-vp_allocation_by_fp` option is only supported by Enginuity 5876 and higher. All Enginuity versions prior to 5876 will display N/A for this option.

---

`-time_to_compress` — Indicates the idle time FAST VP will wait before it compresses devices automatically. The default value is 40 days. The value is in days, the range is 40 to 400 days, or never. Any other value returns an error.

---

**Note:** If your business works on quarterly basis, it is recommended that you set this value to 100 days to account for end of quarter activities.

---

`-fast_compression_rate` — Indicates how aggressive the FAST VP controller works when trying to find candidates for compression. The default value for the FAST compression rate is 5. The values accepted for FAST compression rate are 1 (most aggressive) to 10 (least aggressive).

*Example* To set the FAST controller to `AUTO_APPROVE` mode, the amount of workload sampling for analysis to 200 hours, and the minimum amount of sampling before the recommendation is 100 hours, enter:

```
symfast -sid 207 set -control_parms -mode AUTO_APPROVE -min_perf_period 100
                    -workload_period 200
```

*Example* To set the FAST controller for the maximum concurrent device movement to 8 and the maximum number of devices that can be moved per day to 32, enter:

```
symfast -sid 207 set -control_parms -max_simult_devs 8 -max_devs 32
```

---

**Note:** Metadevice movement is counted as one device.

---

## Displaying FAST control parameters

To display the FAST control parameters, use the following form:

```
symfast [-sid SymmID] [-offline]
list -control_parms
```

---

**Note:** Before using the `-offline` option, the SYMAPI database must be populated with the storage group, tier, and FAST data using the `symcfg sync -fast` command:

---

### Example

To display the control parameters for Symmetrix 432, enter:

```
symfast -sid 432 list -control_parms
```

A display similar to the following displays:

```
Symmetrix ID: 000195700432

Optimizer and FAST Control Parameters:

Data Movement Mode           : User_Approve
Max Simultaneous Device Moves : 8
Max Device Moves Per Day      : 200

Optimizer, FAST and FAST VP Control Parameters:

Min Initial Workload Period(hrs) : 2
Workload Analysis Period(hrs)    : 2

FAST Control Parameters:

Swap Not Visible Devices      : Disabled
Allow Only Swap               : Disabled

FAST VP Control Parameters:

FAST VP Data Movement Mode    : NONE
FAST VP Data Relocation Rate   : 5
Thin Pool Reserved Capacity(%) : 10
VP Allocation By FAST policy   : Enabled
FAST VP Time to Compress      : Never
FAST VP Compression Rate     : 5
```

The features that require Enginuity 5876 Q42012 SR are shown in bold.

These are the possible values for DP data movement mode:

Auto\_Approve — Indicates the FAST controller is in automatic mode.

User\_Approve — Indicates the FAST controller is in user approval mode.

These are the possible values for VP data movement mode:

Auto — Indicates the FAST controller is in automatic mode.

None — Indicates the FAST controller will not perform any data moves on thin devices.

## Displaying FAST plans

**Note:** FAST VP does not display plans for thin devices. However, you can view data movement details in the Optimizer logs (`symoptmz -sid SymmID read -log_type RUNTIME`).

FAST generates optimization plans approximately every hour. FAST plans can contain multiple move/swap suggestions, listed by group. You can query the system at any time to view the current FAST plan.

The FAST controller/Optimizer has the ability to concurrently execute up to two types of plans:

- ◆ FAST DP/traditional Optimizer plan (FAST DP plan)
- ◆ Optimizer manual swap/rollback plan (Optimizer manual plan)

Plans are subdivided into groups that execute serially. Each individual plan has a unique ID.

The `symfast` command does not manage Optimizer manual plans, except to show the history of these activities. Use the following form to display a FAST plan:

```
symfast -sid SymmID
list -plan [-v]
```

**Note:** Both Optimizer plans and FAST plans will display with the `symfast list -plan` command. Check the `Group Attributes` in the output for the origin of the plan.

To list the FAST plan for Symmetrix 234, enter:

```
symfast list -sid 234 -plan
```

Output similar to the following will display:

```

Symmetrix ID          : 000194900234

Plan ID                : 12222009:154359
Plan Type              : Auto Generated
Plan State             : CnfigInProgress
Start Time             : Tue Dec 22 20:30:23 2009
Percent Complete       : 5%
Estimated time to completion : 04:12:30
Number of Groups       : 3

Group 1:
{
  Group Attributes      : Optimizer Generated
  Group State           : InProgress
  Time Started          : Tue Dec 22 20:30:23 2009
  Time Completed        : N/A
  Percent Complete      : 20%
  Estimated time to completion : 00:40:12

Swap Pairs (2)
{
  Source Device          Target Device
  -----
  Dsk                    Dsk
  Sym  Grp Group Name    Prot  Sym  Grp Group Name    Prot

```

```

    0023  004  sata_disks      R5(3+1)  0088  004  sata_disks      R5(3+1)
    0032  004  sata_disks      R1       0058  004  sata_disks      R1
  }
}

Group 2:
{
  Group Attributes      : FAST Generated(Performance)
  Group State           : NotStarted
  Time Started          : N/A
  Time Completed        : N/A
  Percent Complete      : 0%
  Estimated time to completion : 01:35:31

  Swap Pairs (2)
  {
    -----
    Source Device                               Target Device
    -----
    Sym  Tier Name  Prot  Storage  Sym  Tier Name  Prot  Storage
    -----
    0045 PrimeTier  R1    OraSales  00E0 PrimeDBTier R5(3+1) OraSales
    0046 PrimeTier  R1    OraSales  00E1 PrimeDBTier R5(3+1) OraSales
  }
}

Group 3:
{
  Group Attributes      : FAST Generated(Compliance)
  Group State           : NotStarted
  Time Started          : N/A
  Time Completed        : N/A
  Percent Complete      : 0%
  Estimated time to completion : 04:12:30

  Move Devices (6)
  {
    -----
    Source Device                               Target
    -----
    Sym  Tier Name  Prot  Dsk Storage  Tier Name  Prot  Dsk
    -----
    0042 N/A        R6(14+2) 003 OraSales  ArchiveDBTier R5(7+1) 004
    0043 N/A        R6(14+2) 003 OraSales  ArchiveDBTier R5(7+1) 004
    0044 N/A        R6(14+2) 003 OraSales  ArchiveDBTier R5(7+1) 004
    0059 PrimeDBTier R5(3+1) 002 OraSales  WorkDBTier    R5(3+1) 003
    005A PrimeDBTier R5(3+1) 002 OraSales  WorkDBTier    R5(3+1) 003
    005B PrimeDBTier R5(3+1) 002 OraSales  WorkDBTier    R5(3+1) 003
  }
}

```

## FAST plan output description

In the previous display, the `Group Attributes` can have the following values:

**FAST Generated** — Indicates the swap/move plan was generated by the FAST controller.

**Optimizer Generated** — Indicates the swap/move plan was generated by the Symmetrix Optimizer.

In the previous display, the `Group State` can have the following values:

**Done** — The group has finished everything.



`InProgress` — The group is currently running. If running, the `Percent Complete` and the `Estimated time to completion` fields report the progress.

`NotStarted` — The group has not started running yet.

`Failed` — The execution of the entire group has failed and will be retried.

In the previous display, the `Plan State` field can have the following possible values in `USER_APPROVE` mode and will be set to `NA` in `AUTO_APPROVE` mode:

`NotApproved` — Indicates that this is a proposed plan for the FAST controller and the user needs to approve it before the FAST controller can implement the plan.

`ApprovedWithSpecifiedTime` — Indicates that the plan has been approved and scheduled at a specific time.

`ApprovedWithDelay` — Indicates that the plan has been approved and is waiting in a queue to run.

`ApprovedWithConfigParameters` — Indicates that the plan has been approved and will be scheduled as per the time window and control parameter settings.

`InProgress` — Indicates that the plan is currently running.

`Aborting` — Indicates that the plan is currently aborting. This state can occur when the user attempts to decline a running plan.

For FAST DP plans, the plan listing also displays source tier name, source disk group number, and source RAID protection type for device moves. The target disk group name longer displays for device moves, although target disk group number still displays. Time started and completed also displays for each group.

## Approving a FAST plan

When FAST is in `USER_APPROVE` mode, it generates plans approximately every hour, but does not act on them. A FAST plan can be approved or declined. FAST only executes approved plans. Query the system often to be shown the current FAST plan.

---

**Note:** FAST plans cannot be partially approved. Plans are approved or declined as a whole.

---

Use the `symfast -plan approve -id PlanID` command to approve a FAST plan. If you want the plan to start at a specific time, you can add the `-begin_at TimeVal` option to the command.

```
symfast -plan -sid SymmID [-i Interval] [-c Count] [-noprompt]
      approve -id PlanID [-begin_at=TimeVal]
```

To identify the *PlanID*, use the `Plan ID` returned in the `symfast list -plan` output. The `-begin_at` option specifies the specific time to run the FAST plan. If this option is not specified, FAST defaults to the time window definition. The *TimeVal* format is `MMDDYYYY:HHMMSS`.

---

**Note:** Use the `-i` and `-c` options to monitor FAST progress over time.

---

*Example*

To approve the FAST plan 08022009:150115 on Symmetrix 207, and execute this plan during the scheduled time window, enter:

```
symfast -plan -sid 207 approve -id 08022009:150115
```

The FAST plan 08022009:150115 is approved.

---

**Note:** The `approve` command can only be used in `USER_APPROVE` mode.

---

## Declining a FAST plan

To decline a FAST plan, use the following form:

```
symfast -plan -sid SymmID [-i Interval] [-c Count] [-noprompt]
      decline -id PlanID
```

A running plan can be declined. The FAST controller will attempt to abort a running plan; it will complete the current group, and not start any other group of changes.

*Example*

To decline the FAST plan 08022009:150115 on Symmetrix 207, enter:

```
symfast -plan -sid 207 decline -id 08022009:150115
```

The FAST controller deletes the plan. A plan can be declined both before approval and after approval. With Solutions Enabler V7.2, an approved plan can be declined while in progress, but the current in progress group will run to completion.

The `decline` action of `symfast` allows an in-progress FAST standard plan to be aborted, even in `AUTO_APPROVE` mode. Aborting a plan deletes all groups except for the in-progress group. The FAST controller can immediately generate and begin execution of a new plan unless it is disabled or the approval mode is changed to `USER_APPROVE`.

---

**Note:** The `decline` command can be used in either `USER_APPROVE` or `AUTO_APPROVE` mode to abort a running plan.

---

## Displaying FAST information

This section contains list and show displays to use to examine FAST data. Reports about the following topics are explained:

[“FAST activity history” on page 211](#)

[“Audit log messages” on page 213](#)

The next section, [“FAST reports” on page 214](#), provides additional examples of data that can be generated about FAST configurations.

### FAST activity history

**Note:** FAST VP does not show thin device history.

Use the following syntax to display the activity history, or the FAST plan for a Symmetrix array:

```
symfast [-sid SymmID] list -history [-v]
        [-start_date TimeVal] [-end_date TimeVal]
```

Optionally use the `-start_date` and `-end_date` to narrow down the time range.

*Example* To view the FAST activity history for Symmetrix 207, enter:

```
symfast -sid 234 list -history
```

Output similar to the following displays:

**Note:** The following output includes the Optimizer manual device swap history (Group 1).

```
Symmetrix ID          : 000194900234
Number of Groups      : 4

Group 1:
{
  Time Started          : Tue Dec 22 09:30:23 2009
  Time Completed        : Tue Dec 22 09:35:24 2009
  Group Attributes      : Optimizer Manual Swap

  Swap Pairs (2)
  {
    -----
    Source Device          Target Device
    -----
    Dsk
    Sym  Grp Group Name    Prot    Sym  Grp Group Name    Prot
    ---  ---
    0020 004 sata_disks     R5(3+1) 0086 002 flash_disks    R5(3+1)
    0031 004 sata_disks     R1       0055 002 flash_disks    R1
  }
}
```

```
Group 2:
{
  Time Started          : Tue Dec 22 20:30:23 2009
  Time Completed        : Tue Dec 22 20:35:24 2009
  Group Attributes      : Optimizer Generated

  Swap Pairs (2)
```

```

{
    ----- Source Device -----
    Dsk
    Sym  Grp  Group Name      Prot
    ----  ---  -
    0023  004  sata_disks      R5(3+1)
    0032  004  sata_disks      R1
}

}

Group 3:
{
    Time Started           : Tue Dec 22 20:40:50 2009
    Time Completed         : Tue Dec 22 20:45:23 2009
    Attributes              : FAST Generated(Performance)

    Swap Pairs (2)
    {
        ----- Source Device -----
        Dsk
        Sym  Tier Name  Prot  Storage Group Name
        ----  -
        0045 PrimeTier   R1    OraSales
        0046 PrimeTier   R1    OraSales
    }

}

Group 4:
{
    Time Started           : Tue Dec 22 20:50:23 2009
    Time Completed         : Tue Dec 22 21:05:20 2009
    Attributes              : FAST Generated(Compliance)

    Move Devices(6)
    {
        ----- Source Device -----
        Dsk Storage
        Sym  Tier Name  Prot  Grp Group Name
        ----  -
        0042 N/A          R6(14+2) 003 OraSales
        0043 N/A          R6(14+2) 003 OraSales
        0044 N/A          R6(14+2) 003 OraSales
        0059 PrimeDBTier R5(3+1) 002 OraSales
        005A PrimeDBTier R5(3+1) 002 OraSales
        005B PrimeDBTier R5(3+1) 002 OraSales
    }

}

```

The Group Attributes field can have the following possible values:

- **FAST Generated** — Indicates that the swap/move was generated by the FAST controller. May include the logic behind the action, such as Performance, or Compliance (to policy).
- **Optimizer Generated** — Indicates the swap was generated by Optimizer.
- **Optimizer Manual Swap** — Indicates the swap was initiated by the user using the optimizer manual mode.
- **Optimizer Manual Rollback** — Indicates the swap/move was generated as a part of the optimizer roll back activity.

## Audit log messages

The following FAST user actions will have entries in the Symmetrix audit log:

- ◆ Create/delete a tier; add/remove disk groups
- ◆ Create/delete a policy; add/remove tiers; rename a policy; change a policy tier capacity
- ◆ Associate/disassociate a storage group to/from a policy; change the priority of a storage group association
- ◆ Enable/disable the FAST controller; approve/decline a plan
- ◆ Set FAST control parameters

*Example* An example of an audit log entry follows:

```
Record Number      :      181
Records in Seq     :        1
Offset in Seq      :        3
Time               : 02/23/09 15:21:13
Vendor ID          : EMC Corp
Application ID      : SYMTIER
Application Version : 7.1.0
API Library        : SDK
API Version        : X7.1.0 (Edit Level: 1101)
Host Name          : api1182.lss.
OS Name            : LINUX
OS Revision        : 2.6.9-22.0
Client Host        :
Process ID         : 00030023
Task ID           : 00000002
Function Class     : CfgChg
Action Code        : Create
Text               : Starting a Tier 'CREATE' operation for Tier
                    "PrimeDBTier".
                    Options=(FC, RAID6(6+2), Static)
Username           : H:api1182\root
Activity ID        : SEc7be45528f
```

## FAST reports

FAST provides two types of reports about the FAST configuration:

**Compliance report** — Displays compliance information for all storage groups associated with FAST policies, including policies containing VP tiers and policies containing DP tiers. The compliance report examines compliance for each FAST storage group in isolation; contention for tier capacity among storage groups is not accounted for, nor is tier capacity consumed by devices not under FAST control. See [“Compliance report” on page 214](#).

**Technology demand report** — Provides information about this contention for tier resources. See [“Technology demand reports” on page 216](#).

The compliance report and technology demand report together show whether the most efficient use of storage is in place.

### Compliance report

A storage group is FAST-compliant only if all devices in the storage group exist only on the tiers defined in the policy and the percentage capacity of all the tiers occupied by the storage group are within the upper limits of the tier capacities specified in the policy.

The `symfast` command provides options for generating reports about storage groups, policies, and the current logical capacity demand of that association. To generate a report for the demand of a storage group and its associated FAST policy, use the following form:

```
symfast -sid SymmID [-offline]
      list -association [-demand [-sg SgName |
                                -fp_name PolicyName] [-mb]]
```

Where:

`-association` — Displays FAST associations that exist between storage groups and FAST policies.

`-demand` — Displays details about the demand of FAST storage groups on storage tiers.

`-sg` — Displays a report for a specific storage group.

`-fp_name` — Displays a report for a specific FAST policy.

`-mb` — Displays the reported capacities in MBs instead of the default (GB).

---

**Note:** Prior to using the `-offline` option, the SYMAPI database must be populated with tier, storage group, and FAST data using the command `symcfg sync -fast`.

---

For storage groups associated with policies containing DP tiers, the `limit`, `FAST SG Used`, and `Growth` columns will be based on the logical capacity of standard devices in the storage group only. The capacity of any thin devices in the storage group will not be counted.

For storage groups associated with policies containing VP tiers, the `limit`, `FAST SG Used`, and `Growth` columns will be based on the allocated capacity of thin devices in the storage group only. The capacity of any standard devices in the storage group will not be counted, nor will any unallocated logical capacity of the thin devices. The display also indicates

whether the RDF Coordination value is set for each association. For FAST disk group associations, this value is always *N/A*. For FAST VP associations, *Enable* or *Disable* displays.

The following example provides output for the `list -association -demand` command:

**symfast list -sid 432 -association -demand**

Symmetrix Id : 000194900432

Policy Name : HR\_DP  
Storage Group : HR\_DP

Priority : 1  
RDF Coordination : N/A

Tiers (1)

{

Logical Capacities (GB)						
Name	Type	Target Prot	Max SG Percent	Max SG Demand	FAST SG Usage	Growth
HR_EAV	DP	R1	100	111	0	+111
[OutOfPolicy]	N/A	N/A	-	-	111	-111
Total				111	111	

}

Policy Name : HR\_FP  
Storage Group : HR\_SG  
Priority : 3  
RDF Coordination : Enabled

Tiers (1)

{

Logical Capacities (GB)						
Name	Type	Target Prot	Max SG Percent	Max SG Demand	FAST SG Usage	Growth
HR_TIER	VP	R1	100	4	2	+2
Total				4	2	

}

Legend:

Tier Type : DP = Disk Group Provisioning, VP = Virtual Pools

## Compliance report column descriptions

The compliance report columns provide the following information:

[OutOfPolicy] row in the *Name* column — Shows the devices in the storage group that currently do not reside on the tiers defined in the FAST policy.

Max SG Percent column — Shows the limit of the storage group per tier as defined in the FAST policy.

Max SG Demand column — Shows the calculated upper limit, in GB, for the storage group on the tier.

**FAST SG Usage** column — Shows the current occupancy of the storage group in a tier.

**Growth** column — Shows, as per the FAST policy, how much more the storage group can grow on a given tier. This column also indicates compliance. If the potential to grow is negative, the storage group has exceeded the capacity limit for this tier, therefore the storage group is out of compliance.

## Technology demand reports

Technology demand reports show the state of the FAST configuration from the point of view of technology and tiers. The thin demand report presents a Symmetrix-wide snapshot of the current allocations of thin devices under FAST control and of the capacities of thin pools in thin tiers. The report shows the demand the thin devices place on each of the pools if no new allocations were made.

To generate a report for the demand (or details about) a technology type and its tiers, use the following form:

```
symfast [-sid SymmID] [-offline]

list <-tech <EFD | FC | SATA | ALL> -external>
-demand [-v] [-dp | -vp][--allocated]
```

By default, Max SG Demand is calculated using the logical (configured) capacity of thin devices under FAST control. If `--allocated` is specified, Max SG Demand is calculated using the allocated capacity of thin devices under FAST control.

When specifying `-external`, do not use with the `-technology` option.

If neither `-dp` or `-vp` is specified, `-dp` is the default. If the array only contains VP tiers, and the `-vp` option is not specified, the report will show the VP tiers.

**Note:** Prior to using the `-offline` option, the SYMAPI database must be populated with tier, storage group, and FAST data using the command `symcfg sync -fast`.

*Example* The following example shows the command and output for a SATA tier:

```
symfast list -sid 432 -tech SATA -demand -vp
```

```
Symmetrix ID : 000195700432
```

```
Technology : SATA
```

```
VP Tiers (4)
```

```
{
```

	A	Logical Capacities (GB)									
	T										
Tier	T	Target	Tier	Tier	Tier	FAST	SG	FAST	Max	SG	Excess
	R	Prot	Enabled	Free	Used	Usage		Avail	Demand		
HR_test	N	R1	889	883	6		0	0	-		-
PAYRL_test_t*	N	R1	4	4	0		0	0	-		-
PAYRL_tier	F	R1	4	4	0		0	3	8		-5
RECV_366859	N	R1	0	0	0		0	0	-		-
Total			897	891	6		0	3	8		-5

```
}
```

Legend:



ATTR : F = Tier in a FAST policy associated with SG(s)  
 : P = Tier in a FAST policy unassociated with SG(s)  
 : N = Tier not in any FAST policy

The VP demand report displays information only for VP tiers and for storage groups associated with FAST policies containing VP tiers. The `symfast list -external -demand -vp` display adds another block for external tiers. The technology will display N/A for external tiers.

**symfast -sid 432 list -external -demand -vp**

Symmetrix ID : 000194900432

Technology : N/A  
 Disk Location: External

VP Tiers (1)

-----										
		A	Logical Capacities (GB)							
		T	-----							
Tier		T	Tier	Tier	Tier	FAST	FAST	Max	SG	Excess
		R	Enabled	Free	Used	Usage	Avail	Demand		
-----										
external_test	N	Unprot	894	888	6	0	0	-	-	-
Total			894	888	0	0	0	-	-	-
-----										

Legend:

ATTR : F = Tier in a FAST policy associated with SG(s)  
 : P = Tier in a FAST policy unassociated with SG(s)  
 : N = Tier not in any FAST policy

The columns in the previous output example are defined as follows:

**Tier** — Shows names of VP tiers.

**ATTR** — Shows the status of the VP tier. Tiers can have one of 3 possible attributes:

- In a FAST Policy associated with a storage group (F)
- In a FAST Policy or Policies where none of the FAST Policies are associated with a storage group (P)
- Not in any FAST Policy (N)

**Target Prot** — Target protection of the tier.

**Tier Enabled** — Total logical capacity enabled for the tier.

**Tier Free** — The enabled logical capacity minus the used logical capacity, minimum 0.

**Logical Tier Used** — Total pool allocated capacity for all thin pools in the tier. Allocated capacity on all DATA devices will be counted, including DATA devices that are not enabled; therefore Used may be greater than Enabled.

**Logical FAST SG Usage** — Sum of allocated capacity residing on this tier from thin devices in a storage group associated with a FAST Policy containing VP tiers. A separate [OutOfTier] line lists the allocated capacity from such thin devices in a pool of matching technology type that is not included in any VP Tier.

**Logical FAST Available** — If the thin tier is in a FAST policy associated with a storage group, the FAST Available capacity is equivalent to the Tier Free value less the PRC from all thin pools included in the thin tier (with a minimum value of zero) plus the FAST SG Usage. If the tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then this value will be 0.

**Logical Max SG Demand** — Sum of the allocated capacity of all thin devices in a storage group associated with a FAST policy containing this VP tier if the devices were to occupy the full allotted quota (per the limit defined in the FAST policy) of space in the VP tier. Logical Max SG Demand also equals the sum of the values in the Max SG Demand column for all entries for this tier in the compliance report. If the tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then this value is not applicable.

**Logical Excess** — Difference between FAST Available and Logical Max SG Demand. If the thin tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then this value is not applicable.

[Appendix C, “FAST Output Examples,”](#) provides an example of this technology demand report for thin devices using the verbose (-v) option.

# CHAPTER 5

## Federated Tiered Storage

This chapter describes Federated Tiered Storage (FTS) concepts and explains how to configure and manage FTS using the SYMCLI. This chapter covers the following topics:

- ◆ [What is FTS?](#)..... 220
- ◆ [Using eDisks](#) ..... 221
- ◆ [Geometry of encapsulated devices](#)..... 222
- ◆ [Adding an eDisk](#)..... 224
- ◆ [Solutions Enabler FTS support](#)..... 226

## What is FTS?

**Note:** Solutions Enabler V7.4 with Enginuity 5876 and higher supports FTS on Symmetrix VMAX 20K Series and Symmetrix VMAX 40K Series platforms.

Solutions Enabler V7.5 with Enginuity 5876 Q42012SR and higher adds support for the Symmetrix VMAX 10K Series platform.

Federated Tiered Storage (FTS) gives you the ability to attach external storage to a Symmetrix array. Attaching external storage allows you to use physical disk space on existing arrays while gaining access to Symmetrix features such as local replication, remote replication, storage tiering, data management, and data migration. In addition, FTS simplifies the management of federated multi-vendor or EMC storage arrays.

Figure 19 shows the configuration of the network, storage arrays, and application hosts using FTS. For additional details about configuring FTS, refer to the *EMC Symmetrix Federated Tiered Storage (FTS) Technical Notes*.

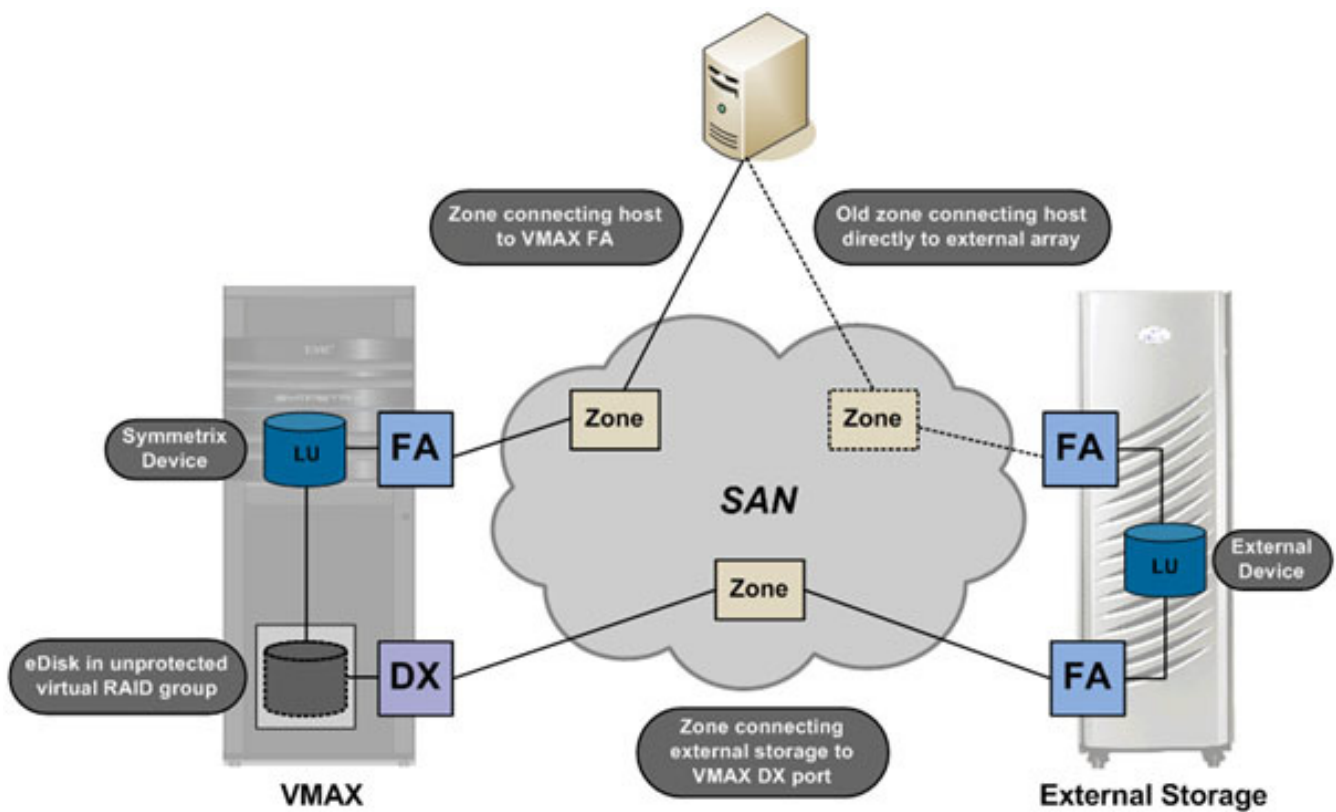


Figure 19 FTS configuration

## Using eDisks

When you attach external storage to a Symmetrix array, FTS virtualizes an external array's SCSI logical units as Symmetrix disks called eDisks. eDisks have two modes of operation:

- ◆ Encapsulation — Allows you to preserve existing data on external arrays and access it through Symmetrix devices. These devices are called encapsulated devices.
- ◆ External Provisioning — Allows you to use external storage as raw capacity for new Symmetrix devices. These devices are called externally provisioned devices. Existing data on the external devices is deleted when they are externally provisioned.

The following restrictions apply to eDisks:

- Can only be unprotected devices. The RAID protection scheme of eDisks is dependent on the external array.
- Cannot be AS400, CKD, or gatekeeper devices.
- Cannot be used as VAULT, SFS, or ACLX devices.

## Encapsulation

Encapsulation has two modes of operation:

- ◆ Encapsulation for disk group provisioning (DP encapsulation) — The eDisk is encapsulated and exported from the Symmetrix array as disk group provisioned devices.
- ◆ Encapsulation for virtual provisioning (VP encapsulation) — The eDisk is encapsulated and exported from the Symmetrix array as thin devices.

In either case, Enginuity automatically creates the necessary Symmetrix devices. If the eDisk is larger than the maximum Symmetrix device size or the configured minimum auto meta size, Enginuity creates multiple Symmetrix devices to account for the full size of the eDisk. These Symmetrix devices are concatenated into a single concatenated meta device to allow access to the complete volume of data available from the eDisk.

## External provisioning

When you virtualize an eDisk for external provisioning, you can then create Symmetrix devices from the external disk group and present the storage to users. You can also use this storage to create a new FAST VP tier.

With Solutions Enabler V7.5 the FTS tier can be associated with a technology type. The technology associated with the FTS tier indicates to the FAST VP controller the expected performance from the tier. This feature enables you to place the FTS tier at the right location for the expected performance of the external tier.

See [“FAST VP overview” on page 175](#) for additional details.

## Geometry of encapsulated devices

Enginuity builds Symmetrix devices based on the Symmetrix cylinder size (fifteen 64 K tracks), so the size of Symmetrix devices will not always match the raw capacity of the eDisk. If the size does not match, Enginuity sets a custom geometry on the encapsulated device. This custom geometry is reported as user defined geometry in Solutions Enabler.

For created meta devices, Enginuity defines the geometry on the meta head, and only the last member can have a size that spans beyond the raw capacity of the eDisk.

## Geometry limited devices

Encapsulated devices that have a Symmetrix cylinder size larger than the reported user-defined geometry size are considered geometry limited. The following restrictions apply to geometry-limited devices:

- Does not support TimeFinder/Snap or TimeFinder/Mirror. Can only be used as source devices for TimeFinder/Clone operations. The rules for operations to larger target devices apply. See the *Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide* for details about copying data from a source device to a larger target device.

If the devices are meta devices, the following requirements must be met:

- The number of meta members on the source device and the target device must be the same.
- For each of the source device and target device meta members, the Symmetrix cylinder size must be the same.

- Can only be used as R1 devices for SRDF migration operations. The rules for operations to larger R2 devices apply. See the *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide* for details about copying data from an R1 device to a larger R2 device. Does not support SRDF/AR.

If the devices are meta devices, the following requirements must be met:

- The number of meta members on the device that contains the R1 and the device that contains the R2 must be the same.
- For each of the meta members on the device that contains the R1 and the device that contains the R2, the Symmetrix cylinder size must be the same.
- Cannot be expanded, dissolved, converted, or used as the target of a VLUN migration.
- Thin devices cannot be unbound or rebound from their pool, and their space cannot be reclaimed.
- Thin pools created during encapsulation cannot have data devices added to them unless the addition is the result of encapsulating external storage.
- Thin pools cannot be added to tiers.

## Encapsulation examples

This section provides examples of geometry-limited encapsulation and encapsulation that is not geometry limited.

### Geometry-limited encapsulation

Assume that you virtualize an eDisk and encapsulate the data for a 22 GB external device with the automatic meta member size on the Symmetrix array configured to 9 GB. The following actions occur:

1. Enginuity creates three 9 GB Symmetrix devices and associates each one with an unprotected, virtual RAID group.
2. Enginuity concatenates the Symmetrix devices into a single concatenated meta device.
3. Enginuity sets the user defined geometry to reflect the encapsulated 22 GB capacity.

This device is considered geometry limited because the Symmetrix device size reported on the meta head (27 GB) is larger than the user defined geometry size of 22 GB.

### Encapsulation that is not geometry limited (meta rules enforced)

Assume that you virtualize an eDisk and encapsulate the data for a 27 GB external device with the automatic meta member size on the Symmetrix array configured to 9 GB. The following actions occur:

1. Enginuity creates three 9 GB Symmetrix devices and associates each one with an unprotected, virtual RAID group.
2. Enginuity concatenates the Symmetrix devices into a single concatenated meta device.
3. Enginuity sets the user defined geometry to reflect the encapsulated 27 GB capacity.

This device is not considered geometry limited because the Symmetrix device size reported on the meta head (27 GB) and the user defined geometry (including the size in bytes) match exactly.

### Encapsulation that is not geometry limited (exact device size enforced)

Assume that you virtualize an eDisk and encapsulate the data for a 22 GB external device, and you override the automatic meta member size configured on the Symmetrix array by specifying the exact device size. The following actions occur:

1. Enginuity creates two 9 GB Symmetrix devices and one 4 GB Symmetrix device and associated each one with an unprotected, virtual RAID group.
2. Enginuity concatenates them into a single concatenated meta device.
3. Enginuity sets the user defined geometry to reflect the exact encapsulated 22 GB capacity.

This device is not considered geometry limited because the Symmetrix device size reported on the meta head (22 GB) and the user defined geometry (including the size in bytes) match exactly.

## Adding an eDisk

When you add an eDisk to a Symmetrix array, you must add it to an external disk group. See [“Managing external disk groups” on page 42](#) for details about creating an external disk group.

You can use the `symsan` command to obtain the WWN of the external LUN that you plan to add. Refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference* for the `symsan` manpage.

The following restrictions apply when adding an eDisk:

- ◆ eDisks can only be associated with one external spindle at a time.
- ◆ The disk group for the eDisk must already exist (or be created first in the same command).
- ◆ With virtual provisioning, either an empty pool or an existing pool composed of external data devices must be provided.

To add an eDisk to an external disk group, create and commit a `symconfigure` command file with the following syntax:

```
add external_disk wwn=wwn to
  disk_group=<DskGrpNum | name:DskGrpName>,
  encapsulate_data=<NO | YES [pool = PoolName]>
  [meta_member_size = n [MB | GB | CYL]]
  [member_size_equal=YES | NO] [dir = director_num];
```

Note the following behavior and requirements when constructing your command file:

- ◆ If the `encapsulate_data` option is set to NO, any data that is currently on the external device is deleted.
- ◆ If you are using virtual provisioning, you must provide the name of an existing thin pool.
- ◆ You can override the auto meta member size configured on the Symmetrix array by specifying the meta member size (`meta_member_size = n [MB | GB | CYL]`).
- ◆ If you do not specify that all of the created Symmetrix devices be the same size (`member_size_equal=YES`), the meta tail will be smaller than the other devices in the meta.

### Examples

To add an eDisk for external provisioning to disk group `exchange_disks1`, create and commit a command file that contains the following command:

```
add external_disk wwn=60000970000194900306533030314341
  to disk_group=exchange_disks1, encapsulate_data=NO;
```

To add an eDisk to disk group `exchange_disks2` and encapsulate the data using DP encapsulation, create and commit a command file that contains the following command:

```
add external_disk wwn=60000970000194900306533030314345
  to disk_group=exchange_disks2, encapsulate_data=YES;
```



To add an eDisk to disk group `exchange_disks2` and encapsulate the data using VP encapsulation (with a thin pool named `eng_pool`), create and commit a command file that contains the following command:

```
add external_disk wwn=60000970000194900306533030314345
    to disk_group=exchange_disks2, encapsulate_data=YES,
    pool=eng_pool;
```

## Removing an eDisk

The following rules apply when removing the eDisk:

- If it was added with encapsulation, the devices on the eDisk must be unmapped and must not be part of a migration, local replication, remote replication, or ORS session. RAID groups and any disk group provisioned devices or DATA devices created during the encapsulation are removed. Thin devices are unbound but they are not removed.
- If it was added for external provisioning, the Symmetrix devices on the eDisk must be deleted before it can be removed.

To remove an eDisk, create and commit a command file that contains the following syntax:

```
remove external_disk <wwn=<wwn> | spid=<SpindleID>>
    [force_remove=<YES | NO>];
```

Where:

`force_remove` — This option allows forcing the removal of an external encapsulated disk when data is still in the Symmetrix cache. Specifically this can occur when the device has iVTOC and write pending tracks.

To remove an eDisk by specifying the eDisks WWN, create and commit a command file that contains a command similar to the following:

*Example*      `remove external_disk wwn=60000970000184700306533030314345;`

To remove an eDisk by using the specifying the spindle ID, create and commit a command file that contains a command similar to the following:

```
remove external_disk spid=2256;
```

## Displaying external disk information

The following sample command and output show the information displayed for external disks:

```
syndisk -sid 647 -spid 1e80 show
```

```
Symmetrix ID           : 000192600647
Director               : DX-16G
Interface              : N/A
Target ID              : N/A
Spindle ID             : 1E80
External WWN           : 60000970000195700233533030333132
External Array ID      : 0001957000233
External Device Name    : 031B
Disk Group Number      : 515
Disk Group Name         : mike
Disk Location           : External
Technology              : N/A
```

```

Speed (RPM)                : N/A
Form Factor                 : N/A
Vendor ID                   : EMC
Product ID                  : SYMMETRIX
Product Revision            : N/A
Serial ID                   : N/A
Disk Blocks                 : 4224000
Block Size                  : 512
Actual Disk Blocks          : 4224000
Total Disk Capacity (MB)    : 2063
Free Disk Capacity (MB)     : 0
Actual Disk Capacity (MB)   : 2063
Rated Disk Capacity (GB)    : N/A
Spare Disk                  : False
Spare Coverage              : N/A
Encapsulated                : True
Disk Service State          : Normal
Hypers (1):
{
#   Vol   Emulation      Dev  Type              Mir Mbr Status      Cap (MB)
-----
  1 N/A    FBA            01CF Ext-Data          1   1   Ready          2062
}

```

## Solutions Enabler FTS support

FTS supports the Solutions Enabler features listed in [Table 16](#):

**Table 16** FTS support across Solutions Enabler features

Feature	Support details
FAST VP	Supports VP pools of externally provisioned DATA devices. Solutions Enabler V7.5 allows you to specify the technology type of the FTS tier. In addition, FAST VP supports up to four tiers when one of the four tiers is the external FTS tier.
Open Replicator	<ul style="list-style-type: none"> <li>Full support for externally provisioned device for ORS, RP, and FLM operations.</li> <li>Support for encapsulated devices that are not geometry limited for only ORS and RP operations.</li> <li>Encapsulated devices are not supported for FLM operations.</li> <li>Encapsulated devices that are geometry limited are not supported for ORS pull operations.</li> </ul>
SRDF	<ul style="list-style-type: none"> <li>All SRDF operations are supported for externally provisioned devices with any RDF personality, (R1, R2, etc.).</li> <li>All SRDF operations are supported for encapsulated devices that are not geometry limited with any RDF personality, (R1, R2, etc.).</li> <li>Encapsulated devices that are geometry limited are supported for SRDF migration operations only. Support is limited to R1 devices. The rules of operation (R2 larger than R1) apply.</li> </ul>
SRDF/SAR	<p>The <code>symreplicate</code> CLI supports both encapsulated and externally provisioned devices, as follows:</p> <ul style="list-style-type: none"> <li>Supports externally provisioned devices.</li> <li>Supports encapsulated devices that are not geometry limited.</li> <li>Does not support encapsulated devices that are geometry limited for any SAR operation.</li> </ul> <p>The <code>symrecover</code> CLI supports both encapsulated and externally provisioned devices, as follows:</p> <ul style="list-style-type: none"> <li>Supports externally provisioned devices.</li> <li>Supports encapsulated devices that are not geometry limited.</li> <li>Does not support encapsulated devices that are geometry limited.</li> </ul>

Table 16 FTS support across Solutions Enabler features

Feature	Support details
SRDF/Star	<ul style="list-style-type: none"> <li>• Supports externally provisioned devices.</li> <li>• Supports encapsulated devices that are not geometry limited.</li> <li>• Does not support encapsulated devices that are geometry limited for any Star operation.</li> </ul>
TimeFinder	<ul style="list-style-type: none"> <li>• Supports externally provisioned devices as source and target devices for TimeFinder/Snap, Clone, and Mirror (TF/Clone Emulation) operations.</li> <li>• Supports encapsulated devices that are not geometry limited as source and target devices for TimeFinder/Clone, and TimeFinder/Mirror (TF/Clone Emulation) operations.</li> <li>• Supports encapsulated devices that are geometry limited only for TimeFinder/Clone operations. These devices can be used as a source devices and the rules for operations to larger target devices apply.</li> </ul>
VLUN migration	Supports both encapsulated and externally provisioned devices, however encapsulated devices cannot be used as targets of a VLUN migration and cannot be migrated using the configured space option.



# CHAPTER 6

## Enhanced Virtual LUN Technology

This chapter explains how to change device protection or improve array performance using Enhanced Virtual LUN Technology and Virtual LUN VP mobility. This chapter covers the following topics:

◆ Overview.....	230
◆ Creating virtual LUN migration sessions .....	235
◆ Migrating to configured space .....	236
◆ Migrating to unconfigured space .....	239
◆ Migrating to a thin pool.....	240
◆ Managing virtual LUN migration sessions .....	242
◆ Viewing the audit log .....	248
◆ Virtual LUN migration examples .....	249

## Overview

Enhanced Virtual LUN Technology allows storage administrators to quickly move an application's data to other storage within the same array. A migration can change the protection level and disk type of FBA, CKD, and IBM i 512-byte D910 devices. A migration can use the storage of existing devices that do not contain live data or unconfigured disk space.

Virtual LUN migration is from source devices to equally-sized target devices, or disks with free space equal to the source devices. A migration will not disrupt hosts or internal applications.

A virtual LUN migration may be initiated in the following circumstances:

- ◆ Traditional data management involves moving data to a lower tier as it ages and becomes less critical to the business. Protection might change as the data ages.
- ◆ Emerging tiered storage configurations include moving data continuously during its lifecycle, either to optimize price/performance or respond to cyclical business changes.

A virtual LUN migration can be performed online or offline.

The `symmigrate` command performs virtual LUN migrations.

### Virtual LUN VP mobility

Virtual LUN VP (virtual pools) mobility allows thin devices to be moved between pools. You can request that, for a given list of devices, the data tracks be moved from their current pool to a target pool. In addition, if the devices themselves are not already bound to the target pool, they will be rebound to the target pool.

To implement this thin-to-thin mobility, the `symmigrate` command provides a target type, `-tgt_pool`, as the destination for a migration. The destination pool is specified with `-pool PoolName`.

To run a thin migration, all input source devices must be thin devices. Thin migrations will support up to 20,000 devices per migration.

### Device pool-level migrations

Solutions Enabler V7.4 offers the capability to migrate allocations from one pool to another pool for a specified set of thin devices.

A pool-level migration of thin devices does not change the binding of that thin device to the target pool, therefore the rebind option must be done manually, if the rebind is desired.

This functionality differs from the full migration of a TDEV in that not all the TDEV allocations will be migrated, just the allocations that are in the source pool. Any allocations in pools other than the source pool will not be migrated to the target in this pool-to-pool migration.

## Migration requirements by features

[Table 17](#) lists the hardware, software, disk, and devices requirements for migrating devices using the virtual LUN technology.

**Table 17** Virtual LUN requirements by features

Enhanced Virtual LUN Technology	Virtual LUN thin-to-thin mobility	Pool-to-pool migrations	VP compression
Symmetrix VMAX Family arrays running Enginuity 5874 and higher	Symmetrix VMAX Family arrays running Enginuity 5875 and higher	Symmetrix VMAX Family arrays running Enginuity 5876 and higher	Symmetrix VMAX Family arrays running Enginuity 5876 Q42012 SR and higher
Solutions Enabler V7.0 and higher	Solutions Enabler V7.2 and higher	Solutions Enabler V7.4 and higher	Solutions Enabler V7.5 and higher
Supported disk types: <ul style="list-style-type: none"> <li>Enterprise flash</li> <li>Fibre Channel</li> <li>SATA</li> </ul>	Supported disk types: <ul style="list-style-type: none"> <li>Enterprise flash</li> <li>Fibre Channel</li> <li>SATA</li> </ul>	Supported disk types: <ul style="list-style-type: none"> <li>Enterprise flash</li> <li>Fibre Channel</li> <li>SATA</li> </ul>	Supported disk types: <ul style="list-style-type: none"> <li>Enterprise flash</li> <li>Fibre Channel</li> <li>SATA</li> </ul>
Supported device types: <ul style="list-style-type: none"> <li>Standard Symmetrix devices, unprotected Symmetrix devices</li> <li>Metadevices</li> <li>FBA, CKD, IBM i 512-byte D910 devices</li> </ul>	Supported device types: <ul style="list-style-type: none"> <li>Standard Symmetrix devices, unprotected Symmetrix devices</li> <li>Thin devices</li> <li>Metadevices</li> <li>FBA, CKD, IBM i 512-byte D910 devices</li> </ul>	Supported device types: <ul style="list-style-type: none"> <li>Thin devices</li> </ul>	Supported device types: <ul style="list-style-type: none"> <li>Thin devices</li> </ul>
Unsupported device types: <ul style="list-style-type: none"> <li>VDEVs (TimeFinder Snap)</li> <li>DATA devices</li> <li>Thin devices</li> <li>Vault devices/SFS/VCM</li> <li>SAVE devices</li> </ul>	Unsupported device types: <ul style="list-style-type: none"> <li>VDEVs (TimeFinder Snap)</li> <li>DATA device</li> <li>Vault devices/SFS/VCM</li> <li>SAVE devices</li> </ul>		Unsupported device types: Encapsulated devices

## Virtual LUN technology restrictions

- ◆ You cannot migrate protected devices to an unprotected protection type. Migrations of unprotected source devices to unprotected target are supported.
- ◆ You cannot migrate to the same protection type (RAID 5 (3+1) to RAID 5 (3+1) and the same disk group number (disk group 1 to disk group 1). The desired protection type or disk group number must be different from what exists on the source devices. This restriction is for disk group provisioned devices.
- ◆ The target devices cannot be in any state where the possibility that data is currently being replicated, for example RDF, clone, or snap devices. These devices are blocked from migration. This restriction is for disk group provisioned devices.
- ◆ You cannot migrate a thin device to a standard device, or a standard device to a thin device.
- ◆ Thin devices must be in the `Bound` state to be migrated. If they are in any other state, an error returns. If the thin device is not already bound to the target pool, it will be bound to that pool after the migration completes.

- ◆ Thin devices that are currently involved in an active migration session are not eligible for transitioning to other thin device states until the migration completes and is terminated.

## VP compression guidelines

The `symmigrate` command allows the migration of thin devices to and from pools that have compression enabled. Table 18 details how compression factors into the thin device migrations:

Table 18 Source and target compression guidelines

	Target compression enabled	Target compression disabled
<b>Source compression enabled</b>	The compressed tracks migrate to the target as compressed tracks. The target pool's utilized space increases by the compressed size and the free space decreases by the compressed size. The source pool's utilized space decreases by the compressed size and the free space increases by the compressed size.	Solutions Enabler does not allow a compressed device to be migrated to a pool with compression disabled. Compressed devices must be uncompressed before they can be migrated.
<b>Source compression disabled</b>	The uncompressed (allocated) tracks migrate to the target pool as uncompressed tracks (allocated). The target pool's utilized space increases by the uncompressed (allocated) size and the free space decreases by the uncompressed (allocated) size. The source pool's utilized space decreases by the uncompressed (allocated) size and the free space increases by the uncompressed (allocated) size.	The uncompressed (allocated) tracks migrate to the target pool. The target pool's utilized space increases by the uncompressed (allocated) size and the free space decreases by the uncompressed (allocated) size. The source pool's utilized space decreases by the uncompressed (allocated) size and the free space increases by the uncompressed (allocated) size.

Prior to migrating any allocations, the total allocations for all relevant thin devices are summed, using the sizes outlined in Table 18, and must be less than the remaining free tracks in the target pool or an error returns and the migration does not take place.

## What's happening inside the array

Solutions Enabler has virtualized the RAID architecture so that RAID 1 and RAID 5 protection mechanisms have been virtualized behind the Symmetrix mirror positions.

This allows multiple, independent RAID groups per Symmetrix logical volume. It also reduces the protection positions required.

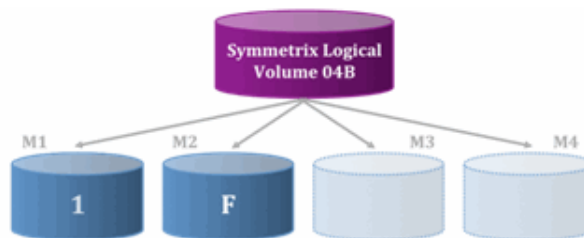
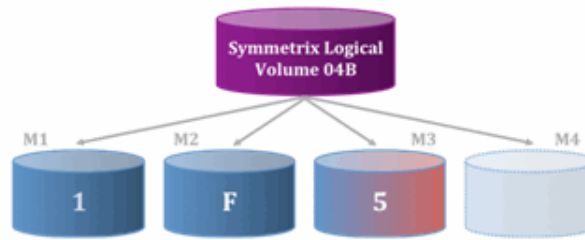


Figure 20 Initial state of RAID 1 device with SRDF



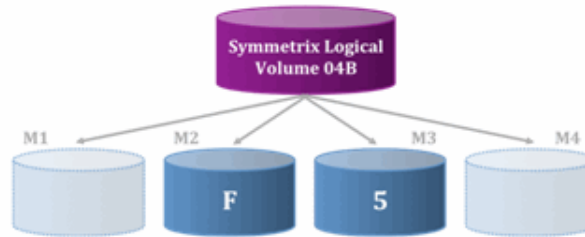
Using virtual LUN technology to migrate a RAID 1 device group device (as shown in [Figure 20](#)) to a RAID 5 device group device, leverages virtual RAID to attach and synchronize the new hypervisors.



**Figure 21** Synchronizing the RAID 5 group

The new target RAID group is associated as an additional mirror to the specified source device. A brief configuration lock is taken for the devices at this time. The system then begins the process of synchronizing the data between the primary and secondary mirrors of the source device, as shown in [Figure 21](#).

Reads and writes are serviced during the migration process.



**Figure 22** RAID 5 group with SRDF

When the data has been fully synchronized on all of the devices associated with a specific migration session, the system automatically promotes the secondary mirror on the source device to be the new primary mirror, as shown in [Figure 22](#).

The following occurs on the target side:

**Configured space** — The original primary mirror of the source devices will become the primary mirror of the target devices.

**Unconfigured space** — The system deletes the original primary mirror and returns the space to the pool of free space on the Symmetrix array.

During the migration the source device will be available for all local and remote replication. The target device will be set by the system as user not ready for the duration of the migration and all operations to set the device to ready will be blocked.

## Source devices

Source devices can be specified by using the following:

- ◆ A device group
- ◆ A storage group
- ◆ A source file (containing a list of source devices or device pairs)

## Target criteria

Target criteria can be specified as configured space or unconfigured space. A target disk group is specified, as well as the target protection type: Unprotected, RAID1, RAID 5 (3+1 or 7+1), or RAID 6 (6+2 or 14+2), unless using pairs in a file. The target protection type and disk group is assumed when using a device file.

The `symmigrate` command now supports a new target type, `-tgt_pool`, as the destination for a thin migration. The destination pool is specified with `-pool PoolName`.

## Using files

### Device file

A device file (*DevFile*) is a text file that contains pairs of source and target devices and one pair per line in the file though the line continuation character of '\' can be used to wrap lines.

---

**Note:** Make sure all the devices in the device file match the type of migration specified.

---

For example:

```
10      20
12      22
14      27
```

### Device file for metadvice

Or for a metadvice:

```
15 28, 29, 30, 31, \
    32, 33, 34, 39
```

A metadvice to metadvice file is also allowed, if the device sizes are equal, and all members are also equal in size and order.

### Source file

A source file (*SrcFile*) is a text file that contains only source devices and one device per line in the file.

For example:

```
9A
9E
9F
```

## FTS support

The `symmigrate` CLI supports both encapsulated and externally provisioned devices.

Encapsulated devices cannot be used as targets of a VLUN migration and cannot be migrated using the configured space option.

Migrations using device pairs for external provisioned devices is fully supported.

## Creating virtual LUN migration sessions

The CLI provides the actions listed in [Table 19](#) to create and manage migration sessions.

**Table 19** Action descriptions for symmigrate command

symmigrate action	Description
establish	Starts a new session and begins the Synchronizing process.
list	Lists all the sessions for a given Symmetrix array, or for all Symmetrix arrays.
query	Queries for the status of the specified session.
terminate	Terminates a session and removes it from the Symmetrix array.
validate	Verifies that the information provided is currently allowed. No changes will be made to the device(s) and the command can optionally output a file containing device pairs (when using configured space).
verify	Verifies that a session is in a specified state.

### Establish

The `symmigrate establish` command creates the migration session and begins the migration of the source devices to the new target location. The target protection type and disk group number must be provided by the user (unless using a device pair file or thin devices). This is the point in the migration process that the mirrors are moved to the source device from the target, if using configured space, or are created if using unconfigured space. The migration of thin devices is similar to the migration of regular devices. You can specify device groups, storage groups, or devices ranges as source thin devices. To run a thin migration, all input source devices must be thin devices, unless the source is a device group or storage group. Thin migrations will support up to 20,000 devices per migration.

Like all other CLI interfaces, the migration control operation will operate on the metahead only. Only the metahead is needed for the control operation; the meta members will be determined by the system. In a configured disk space migration, if the source device is a metadvice, non metadevices can be chosen to match the source device provided that the devices are matched in size and quantity to the meta members. In this case, the output from the `validate` command will be a device-devices list instead of a device-device pair. The source device will be the meta head and the target devices will be a comma separated device list. Alternatively, if there is an available metadvice to use as a target, it can be specified in the standard pair file format.

---

**Note:** CKD striped metadevices can only migrate to unconfigured space, and the target protection type is restricted to RAID-1.

---

### Validate

The `symmigrate validate` command is an optional command, that can be used in two different ways. The command can be used as a simple positive/negative test to determine if a migration will succeed. This will be denoted by the return value of the command. The command can also be used when migrating to configured space to output a file that

contains the exact pairs of devices that will be used for the migration. This output can be used to verify that the command will produce the exact migration that is desired by the user and to modify the resulting file if it is not.

Any `symmigrate` operations performed on a parent SG will be performed on all devices contained by all child SGs of a specified parent SG. Those operations include:

- ◆ `validate`
- ◆ `establish`
- ◆ `terminate`

---

**Note:** The file output is only a listing of what the system chose to do at the time the `validate` command was run. To guarantee that the exact pairs that were returned are used, run the `symmigrate establish` command with the file that was output by the `validate` command.

---

## Migrating to configured space

A migration to configured space can be performed using a device file or by specifying target criteria.

The following sections explain how to perform each type of migration operation.

### Device file

Use the following syntax create a migration session using a device file:

```
symmigrate [-v][-i Interval] [-c Count] [-noprompt]
           -name SessionName
           -file DevFile -sid SymmID
validate
establish
```

For example, if the `input.txt` file contains the following pairs:

```
10      20
12      22
14      27
```

The SYMCLI command would be, as follows:

```
symmigrate validate -sid 123 -f input.txt -name DevMigration
symmigrate establish -sid 123 -f input.txt -name DevMigration
```

### Target criteria

To perform a migration to configured space, and specifying the target criteria, follow this form:

```
symmigrate [-v]
           [-i Interval] [-c Count] [-noprompt]
           -name SessionName
           <-file SrcFile -sid SymmID |
           -sg SgName -sid SymmID |
           -g DgName [-bcv | -tgt]>
           -tgt_config -tgt_dsk_grp DskGrp / name:DskGrpName
           <-tgt_unprotected
           -tgt_raid1
           -tgt_raid5 -tgt_prot <3+1 | 7+1> |
           -tgt_raid6 -tgt_prot <6+2 | 14+2>>
```

```
validate [-outfile OutputFile]
establish
```

Where:

`-name` — Indicates a session name when validating or establishing a session.

The source devices can be identified by one of the following:

- A file, *SrcFile*, which requires the Symmetrix ID
- A storage group (`-sg`), which requires the Symmetrix ID
- A device group, with an optional pointer to the BCV or TGT list

The target criteria is specified as follows:

`-tgt_config` — Tells the array to look for configured space.

`-tgt_dsk_grp` — Identifies the disk group number or name for the migration. The source devices will be in that disk group when the migration is complete.

Protection — The following options specify the protection type that the devices will have when the migration is complete:

`-tgt_unprotected` — Unprotected. This option is only valid if the source devices are unprotected.

`-tgt_raid1` — RAID 1 protection.

`-tgt_raid5` — RAID 5 protection with either `-tgt_prot 3+1` or `7+1`.

`-tgt_raid6` — RAID 6 protection, with either `-tgt_prot 6+2` or `14+2`.

## Verbose error reporting

The `establish` and `validate` migration operations examine all target devices supplied in the user input and return a message for each, indicating whether the device was selected or skipped. The message lists the device number and if skipped, the reason why it was skipped. A device may be skipped when the device is (or has):

- ◆ An invalid configuration
- ◆ A local or remote replication or migration session
- ◆ Host visible
- ◆ Capacity, emulation, protection type, or block size mismatch
- ◆ Reserved

The `symmigrate validate` command can be used to produce a file that will indicate which devices the system identifies as the target *at the time* the `symmigrate validate` command is run.

### Examples

For the following example, the device input file named `test1` contained:

```
0A20 0C90
0A21 0C91
0A22 0C92
0A23 0C93
0A24 0C95
0A25 0C99
```

The following output shows the new messages from a failed `symmmigrate establish` operation for the configured devices shown above in the file `test1`:

```
symmmigrate -file test1 -sid 123 -name test1 establish -nop -v
```

```
`Establish' operation execution is in progress for
the device list in device file `test1'. Please wait...
```

```
STARTING a Migrate `ESTABLISH' operation.
```

```
Source devices:
0A20:0A25 [SELECTED]
```

```
Target devices:
0C90:0C93 [SKIPPED - Local or remote replication or migration]
0C95      [SELECTED]
0C99      [SKIPPED - Host visible]
```

```
The Migrate `ESTABLISH' operation FAILED.
```

```
There are not enough target devices to complete the specified migration
```

For the next example, the following device input file named `test2` was used:

```
0151
```

The following shows the new output from a successful `symmmigrate establish` for a configured migration to a disk group:

```
symmmigrate -file test2 -sid 123 -name test2 -tgt_config
-tgt_dsk_grp 5 -tgt_raid5 -tgt_prot 3+1 -nop -v establish
```

```
`Establish' operation execution is in progress for
the device list in device file `test2'. Please wait...
```

```
STARTING a Migrate `ESTABLISH' operation.
```

```
Source devices:
0151      [SELECTED]
```

```
Target devices:
. . .
0A47      [SKIPPED - Local or remote replication or migration]
0A48      [SKIPPED - Capacity, emulation, prot type or block size mismatch]
0A49:0A4C [SKIPPED - Local or remote replication or migration]
0A4D      [SKIPPED - Capacity, emulation, prot type or block size mismatch]
0A4E      [SELECTED]
0A4F:0A52 [SKIPPED - Local or remote replication or migration]
```

```
Establish Migration.....Started.
Establish Migration.....Done.
```

```
The Migrate `ESTABLISH' operation SUCCEEDED.
```

## Migrating to unconfigured space

This section explains how to perform a virtual LUN migration to unconfigured space.

Use the following form when migrating devices to unconfigured space:

```
symmigrate [-v]
            [-i Interval] [-c Count] [-noprompt]
            -name SessionName
            <-file SrcFile -sid SymmID |
            -sg SgName -sid SymmID |
            -g DgName [-bcv | -tgt]>

            <-tgt_unconfig -tgt_dsk_grp DskGrp / name:DskGrpName

            <-tgt_unprotected
            -tgt_raid1
            -tgt_raid5 -tgt_prot <3+1 | 7+1> |
            -tgt_raid6 -tgt_prot <6+2 | 14+2>>

validate
establish
```

Where:

-name — Indicates a session name when validating or establishing a session.

The source devices can be identified by one of the following:

- A file, *SrcFile*, which requires the Symmetrix ID
- A storage group (-sg), which requires the Symmetrix ID
- A device group, with an optional pointer to the BCV or TGT list

The target criteria is specified as follows:

-tgt\_unconfig — tells the array to look for unconfigured space.

-tgt\_dsk\_grp — identifies the disk group number or name for the migration. The source devices will be in that disk group when the migration is complete.

Protection — the following options specify the protection type that the devices will have when the migration is complete:

-tgt\_unprotected — Unprotected. This option is only valid if the source devices are unprotected.

-tgt\_raid1 — RAID 1 protection.

-tgt\_raid5 — RAID 5 protection with either -tgt\_prot 3+1 or 7+1.

-tgt\_raid6 — RAID 6 protection, with either -tgt\_prot 6+2 or 14+2.

An example of a virtual LUN migration of devices in a device group -tgt list to unconfigured space, with RAID 5 3+1 protection follows:

```
symmigrate validate
-g mydg -tgt
-tgt_raid5 -tgt_prot 3+1 -tgt_dsk_grp 5
-tgt_unconfig -name DGunconfig

symmigrate establish
-g mydg -tgt
-tgt_raid5 -tgt_prot 3+1 -tgt_dsk_grp 5
-tgt_unconfig -name DGunconfig -nop
```

## Error reporting

The `establish` and `validate` migration operations examine all target devices supplied in the user input and return a message stating whether the device was selected or skipped. The message lists the device number and if skipped, the reason why it was skipped. A device may be skipped when the device is (or has):

- ◆ An invalid configuration
- ◆ A local or remote replication or migration session
- ◆ Host visible
- ◆ Capacity, emulation, protection type, or block size mismatch
- ◆ Reserved

*Example* For the following example, the device input file contained:

```
0A33
```

The following output shows the message from a successful `symmigrate establish` for the unconfigured devices:

```
symmigrate -file test1 -sid 123 -name test1 establish -tgt_unconfig -tgt_raid1  
-tgt_dsk_grp 1 -nop -v
```

```
`Establish' operation execution is in progress for  
the device list in device file `test1'. Please wait...
```

```
STARTING a Migrate `ESTABLISH' operation.
```

```
Source devices:
```

```
0A33      [SELECTED]
```

```
Establish Migration.....Started.  
Establish Migration.....Done.
```

```
The Migrate `ESTABLISH' operation SUCCEEDED.
```

## Migrating to a thin pool

Solutions Enabler includes the Virtual LUN VP mobility feature. This feature allows thin devices to be moved between pools. The source thin devices can be specified in a file, device group, or storage group. The target location is specified using a new target type, `-tgt_pool`. The source pool should be removed from FAST control before moving allocations to the target pool.

Use the following form when migrating devices to a thin pool:

```
symmigrate [-v] [-i Interval] [-c Count] [-noprompt] -name SessionName  
  <-file SrcFile -sid SymmID |  
    -sg SgName -sid SymmID |  
    -g DgName [-bcv | -tgt]>  
  -src_pool PoolName  
  -tgt_pool -pool PoolName  
  establish  
  validate
```



Where:

**-src\_pool** — Specifies the name of the pool to be used as the source for a thin migration to a target thin pool. Only allocations within the source pool for the designated set of thin devices will migrate to the target pool.

**-tgt\_pool** — Indicates this is a thin migration. This option must also include the thin pool name (**-pool PoolName**).

The other options and parameters are identical to those defined in [“Migrating to configured space” on page 236](#) and [“Migrating to unconfigured space” on page 239](#), including the verbose error reporting (**-v**).

## Viewing a thin migration

The output display for a thin migration includes the migration type (**Flags**), as shown in the following example:

```
symmigrate -sid 341 -name poolm query -i 60
```

```
Symmetrix ID: 000194900341
```

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	Session Name
0150	N/A	30000	SyncInProg	25	T	poolm
<b>Total</b>						
		Tracks	30000			
		MB(s)	1875			
		Done(%)	25			

### Flags:

(T)ype: T = The session is a thin migration session.  
 C = The session is a configured migration session.  
 U = The session is an unconfigured migration session.

```
Symmetrix ID: 000194900341
```

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	Session Name
0150	N/A	20000	SyncInProg	50	T	poolm
<b>Total</b>						
		Tracks	20000			
		MB(s)	1250			
		Done(%)	50			

```
Copy Rate : 10.4 MB/S
Estimated time to completion : 00:02:00
```

### Flags:

(T)ype: T = The session is a thin migration session.  
 C = The session is a configured migration session.  
 U = The session is an unconfigured migration session.

The **-detail** option provides both the flag and target description to the output display. The **TGT Description** field contains the target pool name if the session is a thin migration or the target protection type if the session is a non-thin migration.

*Examples* The following are two examples of `symmigrate query` with the `-detail` option. The first example shows a query of a thin migration and the second a query of a configured migration:

**`symmigrate -sid 341 -name poolm query -detail`**

Symmetrix ID: 000194900341

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	TGT Dsk Grp	TGT Description	Session Name
0150	N/A	20000	SyncInProg	50	T	01	thinpool	poolm

Total -----  
 Tracks 20000  
 MB(s) 1250  
 Done(%) 50

Flags:

(T): T = The session is a thin migration session.  
 C = The session is a configured migration session.  
 U = The session is an unconfigured migration session.

**`symmigrate -sid 341 -name config query -detail`**

Symmetrix ID: 000194900341

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	TGT Dsk Grp	TGT Description	Session Name
0200	0350	0	Completed	100	C	01	RAID-5 (3+1)	config

Total -----  
 Tracks 0  
 MB(s) 0  
 Done(%) 100

Flags:

(T): T = The session is a thin migration session.  
 C = The session is a configured migration session.  
 U = The session is an unconfigured migration session.

## Managing virtual LUN migration sessions

The `symmigrate` command allows you to list, verify, query, and terminate the migration sessions by using the session name, or by using the session name and specifying the filename, device group, or storage group.

**Note:** If a session has existed for two weeks without being listed or queried, it will automatically be removed, at which point it will not be returned by a `query` or `list` command.

The Symmetrix device lock will be held on the source device(s) only when the `establish` and `terminate` operations are in progress. The lock will be released from the time the migration enters the Sync In Progress state until you terminate the session. When the session is terminated, the lock will be taken again.

A limit of 16 concurrent migration and configuration change sessions can be run for devices in disk group migrations. For thin devices in thin pool migrations, the limit is 128.

## Listing sessions

Use the `symmigrate list` command to display all the existing migration sessions for a specified Symmetrix array, or all Symmetrix arrays, as follows:

```
symmigrate [-v] [-i Interval] [-c Count]
list [-sid SymmID] [-names] [-detail]
```

To display the migration sessions currently on Symmetrix array 432, enter:

```
symmigrate -sid 432 list
```

The following is an example of the output display:

Symmetrix ID: 000194900432

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	Session Name
0210	0325	0	Migrated	100	C	Application1
440	N/A	0	Synchronized	100	U	UNCmigr
06FF	N/A	0	Migrated	100	T	ThinMigr1

```
Total
-----
Tracks      0
MB(s)      0.0
Done(%)    100
```

Flags:

```
(T)ype: T = The session is a thin migration session.
        C = The session is a configured migration session.
        U = The session is an unconfigured migration session.
```

The `-detail` option includes the source and target information for each session in the display, as shown in the following sample output:

Symmetrix ID: 000194900432

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Flags T	TGT Dsk Grp	Description	Session Name
06FF	N/A	0	Migrated	100	T	01	bigpool	ThinMigr1
0210	0325	0	Migrated	100	C	02	fastpool	Application1
440	N/A	0	Synchronized	100	U	01	sparepool	UNCmigr

```
Total
-----
Tracks      0
MB(s)      0.0
Done(%)    100
```

Flags:

```
(T)ype: T = The session is a thin migration session.
        C = The session is a configured migration session.
        U = The session is an unconfigured migration session.
        P = The session is a Virtual Provisioning pool migration session.
```

## Querying sessions

Use the `symmigrate query` command to display the status of a specific migration session, or all migration sessions, as follows:

```
symmigrate [-v] [-i Interval] [-c Count]
           -name SessionName -sid SymmID

query [-detail] [-summary]
```

The `query` command can be used with the session name only, with the addition of the Symmetrix ID.

To display the status of the session named `Migrate1` on Symmetrix 258, enter:

```
symmigrate query -name Migrate1 -sid 258 -detail
```

Symmetrix ID: 000192600258

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	TGT Dsk Grp	TGT Protection	Session Name
0010	N/A	0	Migrated	100	01	RAID-5 (7+1)	Migrate1
0014	N/A	0	Migrated	100	01	RAID-5 (7+1)	Migrate1
-----							
Total		0					
Tracks		0					
MB(s)		0.0					

### Query with -summary option

When the `-summary` option is specified, a summary listing of session states and the number of sessions in each state will be included in the output from each command in order to assist with interpreting the command output.

Used with `symmigrate query`, the `-summary` option replaces the table(s) listing all sessions and their current session states with an abbreviated listing that shows all possible session states and the number of sessions within the scope of the query that are in each state.

The `Total` summary that shows total invalid tracks will be formatted differently when `-summary` is specified.

The estimated time to completion will be shown when:

- ◆ At least one of the `-i` and/or `-c` options is specified.
- ◆ There has been a change in the invalid track counts since the previous iteration.

The following shows the output from `symmigrate query -summary`:

**`symmigrate query -name sess -summary -i 5 c 2`**

Session name: sess

Migration Session State	Count		
-----	-----		
CreateInProg	0		
SyncInProg	2		
Synchronized	0		
MigrateInProg	0		
Migrated	0		
Failed	0		
Invalid	0		
-----	-----		
Total	2		
		Track(s)	MB(s)
		-----	-----
Total Invalid		26863	1678.9

Session name: sess

Migration Session State	Count		
-----	-----		
CreateInProg	0		
SyncInProg	1		
Synchronized	1		
MigrateInProg	0		
Migrated	0		
Failed	0		
Invalid	0		
-----	-----		
Total	2		
		Track(s)	MB(s)
		-----	-----
Total Invalid		11153	697.1
Synchronization Rate	:	109.1 MB/S	
Estimated time to completion	:	00:00:06	

Synchronization rate and estimated time to completion are shown only when `-i` and/or `-c` is specified and the number of invalid tracks has changed since the previous iteration.

## Verifying sessions

The `symmigrate verify` command allows you to determine if a migration session is in a specific state. The syntax follows:

```
symmigrate -sid SymmID -name SessionName [-i Interval] [-c Count]
        verify [-createinprog | -syncinprog | -synchronized
                -migrateinprog | -migrated | -failed | -invalid]
                [-summary]
```

A virtual LUN migration session is complete when it is in the Migrated state.

To verify whether a migration session named `migrate1` is in the migrated state, enter:

```
symmigrate -sid 258 -name migrate1 verify -migrated -i 30
```

The output is similar to the following:

```
NONE of the devices are in the 'Migrated' state.
NONE of the devices are in the 'Migrated' state.
.....
NONE of the devices are in the 'Migrated' state.
ALL of the devices are in the 'Migrated' state.
```

## Verifying sessions with the -summary option

Used with `symmigrate verify`, the `-summary` option provides the same information as provided by the `symmigrate query -summary` command. That identical information precedes the single line summary provided by `symmigrate verify`.

In the following sample command and output, the `symmigrate verify -summary` command verifies the same session as in the previous example of `symmigrate query` using the `-summary` option:

```
symmigrate -sid 341 -name sess verify -failed -summary -i 5 c 2
```

Session name: sess

Migration Session State	Count		
-----	-----		
CreateInProg	0		
SyncInProg	1		
Synchronized	1		
MigrateInProg	0		
Migrated	0		
Failed	0		
Invalid	0		
-----	-----		
Total	2		
		Track(s)	MB(s)
		-----	-----
Total Invalid		26863	1678.9

None of the session(s) with name 'sess' are in 'Failed' state.

Session name: sess

Migration Session State	Count		
-----	-----		
CreateInProg	0		
SyncInProg	1		
Synchronized	1		
MigrateInProg	0		
Migrated	0		
Failed	0		
Invalid	0		
-----	-----		
Total	2		
		Track(s)	MB(s)
		-----	-----
Total Invalid		11153	697.1
Synchronization rate	:	105.6	MB/S
Estimated time to completion	:	00:00:53	

None of the session(s) with name 'sess' are in 'Failed' state.

## Terminating sessions

A migration session remains active until it is terminated. When the status of the devices in the session is Migrated, the session is complete and can be terminated. Thin migrations can be terminated prior to completion.

To terminate a session, use the following form:

```
symmigrate -name SessionName -sid SymmID>
    terminate [-noprompt]
```

## Viewing the audit log

For any control command, the API will log the start and end of the operation as well as any errors that may occur. [Table 20](#) provides a description of the information that is saved in the audit log, based on the target selection type.

**Table 20** Audit log session information

Target selection type	Information in the audit log
Device file	Session name, source devices, target disk group, target protection type, and target devices
Configured/unconfigured space with no specified devices	Session name, source devices, target disk group, and target protection type
A thin pool	Session name, source devices, target protection type, and target pool

For information about the `symaudit` command, refer to the *EMC Solutions Enabler Array Management CLI Product Guide* and the *EMC Solutions Enabler Symmetrix CLI Command Reference*.



## Virtual LUN migration examples

This section provides several virtual LUN migration examples. [Appendix D, “Virtual LUN Migration Example,”](#) provides the step-by-step input and output for a virtual LUN migration session from start to finish.

1. The migration source and target device pairs are in a file, and the target protection type is RAID-5 (3+1).

Device pairs in `input.txt` are:

```
10      20
11      21
12      22
13      23
```

To use this file, enter the following:

```
symmmigrate validate -sid 123 -f input.txt -name Migrate2
```

```
symmmigrate establish -sid 123 -f input.txt -name Migrate2
```

2. You have a storage group `mystoragegrp` that contains devices that you want to migrate to a new protection type but in the same disk group (group number 1), using configured space. The exact target devices that are chosen are irrelevant. For example:

```
symmmigrate validate -sid 123  
-sg mystoragegrp  
-tgt_raid6 -tgt_prot 6+2 -tgt_dsk_grp 1  
-tgt_config -name sgconfig
```

```
symmmigrate establish -sid 123  
-sg mystoragegrp  
-tgt_raid6 -tgt_prot 6+2 -tgt_dsk_grp 1  
-tgt_config -name sgconfig
```

3. You have a device group `mydg` that has a TGT list of devices that needs to be migrated to unconfigured space in disk group 5. The protection type is to remain the same at RAID-5 (3+1). The output file `diskfile` will be created with the chosen physical disks. For example:

```
symmmigrate establish  
-g mydg -tgt  
-tgt_raid5 -tgt_prot 3+1 -tgt_dsk_grp 5  
-tgt_unconfig -name DGunconfig -nop
```

4. To migrate thin devices in a storage group `mystorgrp` to the target pool `fastpool`, enter:

```
symmmigrate -sid 0265 -v -name migrsess -noprompt  
-sg mystorgrp -tgt_pool -pool fastpool establish
```

5. To list all migration sessions established on one Symmetrix array, enter:

```
symmmigrate list -sid 123 -detail
```

6. To monitor a migration session named `mysession`, enter:

```
symmmigrate query -name mysession -sid 123 -i 5 -c 2
```



# CHAPTER 7

## Managing Time Windows

This chapter explains how to add and remove time windows using the `syntw` command. Instructions for converting time windows created in earlier versions of Solutions Enabler are also included. The chapter covers the following topics:

◆ Managing time windows .....	252
◆ Converting time windows .....	254
◆ Adding time windows.....	256
◆ Removing time windows.....	257
◆ Displaying time windows.....	258
◆ Time window examples .....	261

# Managing time windows

## IMPORTANT

This feature requires Enginuity 5875.235.172 or higher.

Time windows are used by FAST, FAST VP, and Symmetrix Optimizer to collect performance statistics and execute data movement within the array. Solutions Enabler V7.3 introduced a new command, `symtw`, to manage time windows. This command improves on the process and formatting of time windows (previously managed with `symoptmz set time_window`).

You can continue to use the `symoptmz` command, or you can convert previously defined time windows to the new format using the `symtw convert` command. However, you cannot use both commands. In addition, the `symtw` command is not available until a `symtw convert` operation executes on a Symmetrix array.

**Note:** After using the `symtw` command, the `symoptmz` command for setting time windows is no longer supported.

There are three different types of time windows:

- ◆ Data movement for disk group provisioned devices (`-dp`).
- ◆ Data movement for virtually provisioned devices (`-vp`).
- ◆ Performance time windows which control the collection of statistics.

In addition, a defined time window needs to be specified as either *inclusive*, which allows the operation to be executed repetitively, or *exclusive* which prevents the operation for a future specific date and time.

The new time window definition includes the following features:

- ◆ All inclusive time windows are similar to the weekly by day time window definitions without the start and end date.
- ◆ The inclusive time windows are defined by using one or more days of the week and the start/end time to be applied to each day. The start and end time are in 30 minute increments from 00:00 to 24:00. The time 00:00 represents midnight AM and 24:00 represents midnight PM.
- ◆ The exclusive time windows can be defined for a period of time with the start date/time and end date/time. The start and end time are in 30 minutes increments from 00:00 to 24:00. The exclusive time window will have the highest priority and it will override any time windows that have been defined during those time period.
- ◆ The time window definitions stored on Symmetrix database are in GMT time. Solutions Enabler V7.3.1 has added the option in the API to display the host local time when adding, removing, or querying the time windows.
- ◆ Any newly added time windows will not replace the current time windows. They will be added on top of the current time windows. The remove operation will allow the user to remove any specified time windows.
- ◆ Any expired exclusive time windows will be deleted whenever the time window database is updated.

- ◆ The `symtw` command supports a `convert` command to translate any legacy time window definitions defined previously using the `symoptmz` command to the enhanced `symtw` time window definitions. After the execution of the `symtw convert` command, time window definition and reporting using the `symoptmz` command will be blocked and all time window management and reporting must be performed using the `symtw` command.
- ◆ There are three system default time windows and the FAST VP controller will collect performance statistics and execute data movement only after a `symtw convert` operation by the user.
- ◆ Symmetrix VMAX 10K/VMAXe arrays only support `symtw` for defining FAST VP time windows. There are predefined performance time windows. They can optionally be removed using the `symtw remove` or `symtw rsmall` commands.

This chapter provides the following information about time windows:

[“Converting time windows” on page 254](#)

[“Adding time windows” on page 256](#)

[“Removing time windows” on page 257](#)

[“Displaying time windows” on page 258](#)

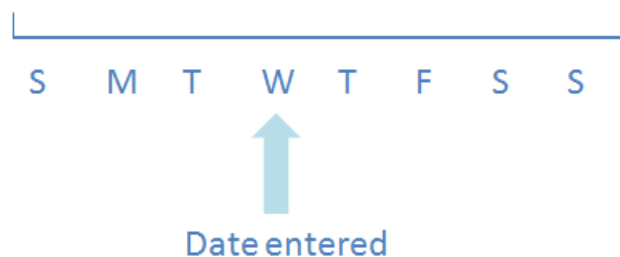
[“Time window examples” on page 261](#)

## Time window restrictions and guidelines

The following guidelines apply to the `symtw` type time window:

- ◆ For the inclusive time windows, the start time and end time cannot be extended to the previous or next day. Any time windows that extend to the next day need to be defined by using a separate `symtw add` commands with the same type.
- ◆ For the exclusive time windows, the time window start date/time and end date/time must be specified and the dates must in the future time.
- ◆ For both inclusive and exclusive time windows, the valid start and end time values for the days are from 00:00 to 24:00 in 30 minutes increments. The time 00:00 represents midnight AM and 24:00 represents midnight PM.

## Time window weekly format



**Figure 23** Time window weekly format

Inclusive time windows *always* utilize a Sunday to Saturday week. Therefore, if you enter a command, such as `symtw convert` with Wednesday's date, as shown in [Figure 23](#), the system begins the conversion process beginning with the previous Sunday and through the seven days to the next Saturday.

Any time windows created in the previous week are deleted, and any time windows in the future after Saturday are deleted. The only exception is a legacy 'exclude' window for one time in the future.

## Converting time windows

The `symtw convert` command converts any legacy time window definitions defined through the `symoptmz` command to the enhanced `symtw` time window format.

You have the option to choose the start date when converting the time windows. When the date is specified in the command line, the enhanced time window definitions are created based on the composite time windows generated from the 7 days starting from the specified date. The converting date will be the current day if the user does not provide the exact date information.

### Conversion guidelines

Follow these guidelines when converting legacy time windows:

- ◆ You must execute the `convert` command before you can use the `symtw` command to manage and report on time windows. All `symtw` commands will return an error if the legacy time window definition format still exists on the Symmetrix array.
- ◆ When the `symoptmz` time window definitions cannot be translated to the new `symtw` time window definitions, an error returns. The `-force` option can be specified to force the best effort conversion of the time windows.
- ◆ The `symoptmz` command is not supported after a successful `symtw convert` operation creates the enhanced time window format on the Symmetrix array.
- ◆ The `symtw` command is not supported if the time window database has been initialized and the format is in the legacy `symoptmz` time window format. When the time window database is uninitialized, the `convert` command is also required and it will set the database to the enhanced time window format.
- ◆ Only the time windows defined during the 7 days including the start date will be converted. All inclusive time windows defined before or after this period of time will be deleted.
- ◆ Exclusive time windows previously defined for a future date will convert to the enhanced time window format only if it is defined as `ONCE`.
- ◆ The start and end time of the time windows will be rounded to the nearest half hour time. The start time will be rounded down to the previous half hour and the end time will be rounded up to the next half hour.

## Conversion examples

The following command will convert the inclusion legacy time window definitions, defined from Feb 6, 2011 through Feb 12, 2011, to the enhanced time window format. In addition all 'once' exclusion legacy time window definitions from Feb 6, 2011 and later will be converted to the enhanced time window format.

The following examples will display the output of `symtw convert` given the legacy time windows, assuming they are in file `legacy_windows` (the contents shown below), and the `symptomtz -sid 432 -f legacy_windows commit` command was executed:

### Example legacy\_windows file contents

```
set time_window id=MyPerf type=perf flag=include period=once
  starting=02202010:000000 ending=02202012:000000;
set time_window id=MyPerf1 type=perf flag=exclude period=WEEKLY
  starting=11042010:200000 ending=12302010:210000 days=WED_START FRI
  start_time=19:00 end_time=20:00;
set time_window id=MyDpDataMove type=swap provisioning=dp flag=include
  period=WEEKLY_BY_DAY starting=11042010:200000 ending=12302012:210000 days=SUN
  MON TUE THUR start_time=19:00 end_time=2:00;
set time_window id=MyDpDataMoveExclude type=swap provisioning=dp flag=exclude
  period=ONCE starting=02112011:120000 ending=02112011:131500;
set time_window id=MyDpDataMoveExclude1 type=swap provisioning=dp flag=exclude
  period=ONCE starting=11242010:200000 ending=11252010:210000;
set time_window id=MyDpDataMoveExclude2 type=swap provisioning=dp flag=exclude
  period=ONCE starting=02142011:200000 ending=02142011:210000;
set time_window id=MyVpDataMove type=swap provisioning=vp flag=include
  period=WEEKLY starting=11042010:200000 ending=12302012:210000 days=WED_START FRI
  start_time=19:00 end_time=20:00;
set time_window id=MyVpDataMove2 type=swap provisioning=vp flag=include
  period=WEEKLY starting=02112011:200000 ending=12302012:210000 days=WED_START FRI
  start_time=19:00 end_time=20:00;
set time_window id=MyVpDataMoveExclude type=swap provisioning=vp flag=exclude
  period=WEEKLY starting=03042011:200000 ending=12302011:210000 days=WED_START
  FRI;
```

In this conversion example the `-force` option is added for a “best effort” conversion:

```
symtw convert -sid 432 -force
```

```
Execute conversion of legacy time windows for Symmetrix 000000000432
```

```
Evaluating DP Move Time Window 'The Default Time Window': Convertible
Evaluating VP Move Time Window 'The Default Time Window': Convertible
Evaluating Performance Time Window 'The Default Time Window': Convertible
Evaluating Performance Time Window 'MyPerf ': Convertible
Evaluating Performance Time Window 'MyPerf1': Expired
Evaluating DP Move Time Window 'MyDpDataMove': Convertible
Evaluating DP Move Time Window 'MyDpDataMoveExclude': Convertible
Evaluating DP Move Time Window 'MyDpDataMoveExclude1': Expired
Evaluating DP Move Time Window 'MyDpDataMoveExclude2': Convertible
Evaluating VP Move Time Window 'MyVpDataMove': Convertible
Evaluating VP Move Time Window 'MyVpDataMove2': Partially Convertible - End of
  period exceeds enhanced time window end date
Evaluating VP Move Time Window 'MyVpDataMoveExclude': Not-Convertible - Weekly
  exclude starts after enhanced time window end date.
```

```
Converting DP Move Time Window 'The Default Time Window': Done
Converting VP Move Time Window 'The Default Time Window': Done
Converting Performance Time Window 'The Default Time Window': Done
Converting Performance Time Window 'MyPerf ': Done
Converting Performance Time Window 'MyPerf1': Skipped
```

```

Converting DP Move Time Window 'MyDpDataMove': Done
Converting DP Move Time Window 'MyDpDataMoveExclude': Done
Converting DP Move Time Window 'MyDpDataMoveExclude1': Skipped
Converting DP Move Time Window 'MyDpDataMoveExclude2': Done
Converting VP Move Time Window 'MyVpDataMove': Done
Converting VP Move Time Window 'MyVpDataMove2': Done
Converting VP Move Time Window 'MyVpDataMoveExclude': Skipped

```

Conversion of legacy Time Windows successfully completed.

## Adding time windows

Use the following syntax to create a time window:

```

symtw -sid SymmID -inclusive [-noprompt]
      -type <move_dp | move_vp | perf | all>
      add -days DayList -start_time Time -end_time Time

symtw -sid SymmID -exclusive [-noprompt]
      -type <move_dp | move_vp | perf | all>
      add -start_day DateTime -end_day DateTime

```

Where:

**-inclusive** — Specifies that you want to create this time window.

**-exclusive** — Specifies that you want the system to exclude these times from any time window.

**-noprompt** — Requests that no prompts are returned after the command is entered. The default is to prompt the user for confirmation.

**-type** — Indicates the type of time window, as follows:

**move\_dp** — Move data on disk provisioned devices.  
**move\_vp** — Move data on virtually provisioned devices.  
**perf** — Gather performance data.  
**all** — Create all three types of time windows.

**Daylist** — Specifies the days for the time window. Possible values are: Any comma-separated combination of MON, TUE, WED, THU, FRI, SAT, and SUN.

**Time** — Specifies the time of day in the format of HH:MM. Possible values are 00:00 to 24:00 for each day in 30 minute increments. The time 00:00 represents midnight AM, and 24:00 represents midnight PM.

**DateTime** — Specifies the date and time in the format of MMDDYYYY:HHMM. The time of day is in 30 minute increments. The valid values for minutes are 0 and 30.

---

Hint: Inclusive time windows only use the *DayList* and *Time* parameters. Exclusive time windows only use the *DateTime* parameter.

---

### *Inclusive time window example*

In the following examples, two *inclusive* time windows are added:

```

symtw -sid 397 add -type MOVE_DP -inclusive
      -days Mon,Tue,Wed,Thu,Fri
      -start_time 18:00 -end_time 24:00

symtw -sid 397 add -type MOVE_VP -inclusive
      -days Mon,Tue,Wed,Thu,Fri
      -start_time 08:00 -end_time 17:30

```



*Exclusive time window example*

In the following examples, two *exclusive* time window are added:

```
symtw -sid 397 add -type MOVE_DP -exclusive
      -start_day 11042011:0000 -end_day 11042011:2300

symtw -sid 397 add -type MOVE_DP -exclusive
      -start_day 12252011:0000 -end_day 12262011:2400
```

## Removing time windows

Use the `symtw remove` command to execute the following operations:

- ◆ Remove the specified inclusive time windows using the `-type`, `-days`, `-start_time` and `-end_time` options.
- ◆ Remove the specified exclusive time window with `-type`, `-start_date` and `-end_date` options.

This command does not require an exact match between the existing time windows and the input time windows from the command when removing the time windows.

Use the following syntax for removing time windows:

```
symtw -sid SymmID -inclusive [-noprompt]
      -type <move_dp | move_vp | perf | all>

      remove -days DayList -start_time Time -end_time Time

symtw -sid SymmID -exclusive [-noprompt]
      -type <move_dp | move_vp | perf | all>

      remove -start_day DateTime -end_day DateTime
```

These options are defined in [“Adding time windows” on page 256](#).

*Inclusive time window example*

The following example removes the time windows from 17:30 to 18:30 on Monday, Tuesday, Wednesday, Thursday and Friday. In this example, the existing time windows are defined as Monday to Friday and the starting and ending time are from 18:00 to 24:00 for each day.

```
symtw -sid 397 remove -type MOVE_DP -inclusive
      -days Mon,Tue,Wed,Thu,Fri
      -start_time 17:30 -end_time 18:30
```

After the execution of the `remove` command, the new time windows will be Monday to Friday and the starting and ending times are from 18:30 to 24:00 for each day.

*Inclusive time window example*

The following example removes the exclusive time window:

```
symtw -sid 397 remove -type MOVE_DP -exclusive
      -start_day 11042011:0000 -end_day 11042011:0800
```

## Removing all time windows

The `symtw rmall` command allows you to clear time windows, as follows:

- ◆ Remove all inclusive time windows of one type or more with the `-type` and `-inclusive` options.
- ◆ Remove all exclusive time windows of one type or more with the `-type` and `-exclusive` options.

- ◆ Remove all exclusive and inclusive time windows of one type or more with the `-type`, `-exclusive`, and `-inclusive` options.

### *Removing all time windows example*

The following command removes all DP inclusive time windows:

```
symtw -sid 397 rmall -type MOVE_DP -inclusive
```

The following command removes all time window definitions:

```
symtw -sid 397 rmall -type ALL -exclusive -inclusive
```

## Displaying time windows

To display the defined time windows, use the `symtw list` command. Use the `-type` option to filter the display for only the requested window types. The list command provides the following options:

```
symtw [-sid SymmID] [-offline]

list [-type <move_dp | move_vp | perf>]
list -summary [-date Date]
```

Where:

- `-type` — Displays only the specified type.
- `-summary` — Displays a calendar view of all defined time windows.
- `-date` — Displays the defined time window for the specified date. The date format is: MMDDYYYY

The following example shows the time windows defined for Symmetrix array 397:

```
symtw -sid 397 list
```

```
Symmetrix ID: 000194900397
```

```
DP Data Movement Time Windows
```

```
Sunday      : None
Monday      : 18:30 - 24:00
Tuesday     : 18:30 - 24:00
Wednesday   : 18:30 - 24:00
Thursday    : 18:30 - 24:00
Friday      : 18:30 - 24:00
Saturday    : None
```

```
Exclusive Time Windows (2)
```

```
{
  Fri Nov  4 08:00:00 2011 - Fri Nov  4 23:00:00 2011
  Fri Dec 25 00:00:00 2011 - Sat Dec 26 24:00:00 2011
}
```

```
VP Data Movement Time Windows
```

```
Sunday      : None
Monday      : 08:00 - 17:30
Tuesday     : 08:00 - 17:30
Wednesday   : 08:00 - 17:30
Thursday    : 08:00 - 17:30
Friday      : 08:00 - 17:30
Saturday    : None
```

```
Exclusive Time Windows (0)
```

## Performance Time Windows

```

Sunday       : 07:00 - 18:00
Monday       : 07:00 - 18:00
Tuesday      : 07:00 - 18:00
Wednesday    : 07:00 - 18:00
Thursday     : 07:00 - 18:00
Friday       : 07:00 - 18:00
Saturday     : 07:00 - 18:00

```

Exclusive Time Windows (0)

## Using the -summary option

Use the `symtw list -summary` command to show a summary of the time window definitions for the current week starting from Sunday. Any exclusive time window that overrides the defined time windows during the current time of period also display.

```
symtw list -sid 397 -summary
```

Symmetrix ID: 0001949000397

Time Window Summary							
	SUN	MON	TUE	WED	THU	FRI	SAT
	D V P	D V P	D V P	D V P	D V P	D V P	D V P
00:00 - 00:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
00:30 - 01:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
01:00 - 01:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
01:30 - 02:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
02:00 - 02:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
02:30 - 03:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
03:00 - 03:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
03:30 - 04:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
04:00 - 04:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
04:30 - 05:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
05:00 - 05:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
05:30 - 06:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
06:00 - 06:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .
06:30 - 07:00	. . .	. . .	. . .	. . .	. . .	. . .	. . .
07:00 - 07:30	. . P	. . P	. . P	. . P	. . P	. . P	. . P
07:30 - 08:00	. . P	. . P	. . P	. . P	. . P	. . P	. . P
08:00 - 08:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
08:30 - 09:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
09:00 - 09:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
09:30 - 10:00	. . P	. V P	. V P	. V P	-> . V P <-	. V P	. . P
10:00 - 10:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
10:30 - 11:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
11:00 - 11:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
11:30 - 12:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
12:00 - 12:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
12:30 - 13:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
13:00 - 13:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
13:30 - 14:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
14:00 - 14:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
14:30 - 15:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
15:00 - 15:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
15:30 - 16:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
16:00 - 16:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
16:30 - 17:00	. . P	. V P	. V P	. V P	. V P	. V P	. . P
17:00 - 17:30	. . P	. V P	. V P	. V P	. V P	. V P	. . P
17:30 - 18:00	. . P	. . P	. . P	. . P	. . .	. . P	. . P
18:00 - 18:30	. . .	. . .	. . .	. . .	. . .	. . .	. . .

18:30 - 19:00	. . .	D . .	D . .	D . .	D . .	E . .	. . .
19:00 - 19:30	. . .	D . .	D . .	D . .	D . .	E . .	. . .
19:30 - 20:00	. . .	D . .	D . .	D . .	D . .	E . .	. . .
20:00 - 20:30	. . .	D . .	D . .	D . .	D . .	E . .	. . .
20:30 - 21:00	. . .	D . .	D . .	D . .	D . .	E . .	. . .
21:00 - 21:30	. . .	D . .	D . .	D . .	D . .	E . .	. . .
21:30 - 22:00	. . .	D . .	D . .	D . .	D . .	E . .	. . .
22:00 - 22:30	. . .	D . .	D . .	D . .	D . .	E . .	. . .
22:30 - 23:00	. . .	D . .	D . .	D . .	D . .	E . .	. . .
23:00 - 23:30	. . .	D . .	D . .	D . .	D . .	D . .	. . .
23:30 - 24:00	. . .	D . .	D . .	D . .	D . .	D . .	. . .
	-----	-----	-----	-----	-----	-----	-----

Legend: D = Disk Group Provisioning Movement Time Window  
 V = Virtual Provisioning Movement Time Window  
 P = Performance Time Window  
 E = Time Windows Overridden by the Exclusive Time Windows

The time slot that is marked with -> and <- denotes the time when the `list` command was executed.

The `symcfg list` display indicates the type of time window format, as shown in the next example command and output:

**`symcfg list -sid 397 -v`**

Symmetrix ID: 0001949000397  
 Time Zone : EST

```

Product Model                : VMAX-1
Symmetrix ID                 : 000194900397

Microcode Version (Number)   : 5875 (16F30000)
Microcode Registered Build   : 0
Microcode Date               : 11.19.2010

. . .

IPSec Status                 : Pass Thru
Allow spare in mirror 4 position : Disabled
Disks Service                : Normal
Symmetrix Data Encryption    : Disabled
Time Window Definition Format : Enhanced

Parity Raid Configuration    : N/A
Raid-5 Configuration         : RAID-5 (3+1 and 7+1)
Raid-6 Configuration         : RAID-6 (6+2 and 14+2)
PAV Mode                     : DynamicStandardPAV
PAV Alias Limit              : 31

SRDF/A Maximum Host Throttle (Secs) : 0
SRDF/A Maximum Cache Usage (Percent) : 75

Auto Meta                    : Enabled
Minimum Auto Meta Size       : 2000
Auto Meta Member Size        : 1500
Auto Meta Configuration      : Concatenated
  
```

In the previous output, the **Time Window Definition Format** will be labeled **Legacy** if the time windows were previously created using the `symoptmz` command.

For more information about the `symcfg` command, refer to the *EMC Solutions Enabler Array Management Product Guide*.

## Time window examples

This section provides some additional examples of using the new time window command, `symtw`:

In the example commands below, DP time windows will be created that encompass 00:00 to 09:00 and 12:30 to 24:00 on Saturday as well as the entire day on Sunday. This will be accomplished by adding a window that covers all 24 hours in Saturday and Sunday and then removing the period from 9:00 to 12:30 on Saturday:

```
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Sun,Sat -start_time 00:00 -end_time 24:00  
  
symtw -sid 397 remove -type MOVE_DP -inclusive  
      -days Sat -start_time 09:00 -end_time 12:30
```

Alternatively, the same windows could have been defined using three add commands as shown below:

```
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Sat -start_time 00:00 -end_time 09:00  
  
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Sat -start_time 12:30 -end_time 24:00  
  
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Sun -start_time 00:00 -end_time 24:00
```

The following add command will define a set of DP time windows on Monday, Tuesday, Wednesday, Thursday and Friday from midnight to 6 AM:

```
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 00:00 -end_time 06:00
```

The following commands will define a set of DP time windows on Monday, Tuesday, Wednesday, Thursday and Friday from noon to 2 PM and 4PM to midnight. This can be accomplished in 2 ways either by using 2 add commands or by using a combination of an add command and a remove command:

```
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 12:00 -end_time 14:00  
  
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 16:00 -end_time 24:00  
or
```

```
symtw -sid 397 add -type MOVE_DP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 12:00 -end_time 24:00  
  
symtw -sid 397 remove -type MOVE_DP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 14:00 -end_time 16:00
```

The following add commands will define a set of VP time windows on Monday, Tuesday, Wednesday, Thursday and Friday from 8 AM to midnight:

```
symtw -sid 397 add -type MOVE_VP -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 08:00 -end_time 24:00
```

The following remove commands will modify the VP time windows that have been defined in the previous add command. The first remove command will modify the previously created VP time window to end at 5:30 PM on Monday, Wednesday, and Friday only instead of 24:00. The second remove command will modify the previously created VP time window to end at 8:30 PM on Tuesday and Thursday only instead of 24:00:

```
symtw -sid 397 remove -type MOVE_VP -inclusive  
      -days Mon,Wed,Fri -start_time 17:30 -end_time 24:00
```

```
symtw -sid 397 remove -type MOVE_VP -inclusive  
      -days Tue,Thu -start_time 20:30 -end_time 24:00
```

The following add command will add a VP time window from 09:00 to 12:30 on Saturday:

```
symtw -sid 397 add -type MOVE_VP -inclusive  
      -days Sat -start_time 09:00 -end_time 12:30
```

The following add commands will add PERF time windows for weekdays and an additional window for Saturday:

```
symtw -sid 397 add -type PERF -inclusive  
      -days Mon,Tue,Wed,Thu,Fri -start_time 07:00 -end_time 22:00
```

```
symtw -sid 397 add -type PERF -inclusive  
      -days Sat -start_time 08:00 -end_time 13:00
```

The following output example shows the time windows that have been defined in previous commands:

```
symtw -sid 397 list
```

```
Symmetrix ID: 000194900397
```

#### DP Data Movement Time Windows

```

Sunday      : 00:00 - 24:00
Monday      : 00:00 - 06:00
              12:00 - 14:00
              16:00 - 24:00
Tuesday     : 00:00 - 06:00
              12:00 - 14:00
              16:00 - 24:00
Wednesday   : 00:00 - 06:00
              12:00 - 14:00
              16:00 - 24:00
Thursday    : 00:00 - 06:00
              12:00 - 14:00
              16:00 - 24:00
Friday      : 00:00 - 06:00
              12:00 - 14:00
              16:00 - 24:00
Saturday    : 00:00 - 09:00
              12:30 - 24:00

```

```
Exclusive Time Windows (0)
```

#### VP Data Movement Time Windows

```

Sunday      : None
Monday      : 08:00 - 17:30
Tuesday     : 08:00 - 20:30
Wednesday   : 08:00 - 17:30
Thursday    : 08:00 - 20:30
Friday      : 08:00 - 17:30

```

Saturday : 09:00 - 12:30

Exclusive Time Windows (0)

Performance Time Windows

Sunday : None

Monday : 07:00 - 22:00

Tuesday : 07:00 - 22:00

Wednesday : 07:00 - 22:00

Thursday : 07:00 - 22:00

Friday : 07:00 - 22:00

Saturday : 08:00 - 13:00

Exclusive Time Windows (0)





# CHAPTER 8

## Managing Quality of Service

This chapter describes the QoS concepts and how to manage the Quality of Service (QoS) on devices in your storage environment and how to manage dynamic cache partitions in your Symmetrix array by using the SYMCLI `symqos` command. The chapter covers the following topics:

◆ Overview.....	266
◆ Setting copy pace and priority .....	266
◆ Setting host I/O priority .....	270
◆ Setting and resetting LRU parameters .....	271
◆ Managing dynamic cache partitions .....	275
◆ QoS examples.....	280

## Overview

Quality of Service (QoS) allows you more flexibility in managing your Symmetrix system's performance. By increasing the response time for specific copy operations on selected devices or groups, you can increase the overall performance of the other Symmetrix devices.

The QoS (Quality of Service) SYMCLI feature allows you to adjust the data transfer (copy) pace on specified devices, or devices in a device group or storage group, of certain operations.

The contention for cache access can be quality of service managed by the least recently used (LRU) ring partitions in the Symmetrix cache.

With Solutions Enabler, you can control the priority service time of devices and create and manage cache partitions for different device groupings.

The `symqos` command provides Quality of Service controls on specified devices. [Table 21](#) lists the features of the QoS command.

**Table 21** `symqos` command descriptions

Command	Description
<code>symqos</code>	<p>Invokes quality of service controls on certain devices. Possible operations are:</p> <ul style="list-style-type: none"> <li>• Sets the copy pace for device groups and storage groups.</li> <li>• Sets priority service time for a range of devices, a device group, or a storage group.</li> <li>• Enables/disables priority service time for devices.</li> <li>• Queries the QoS copy priorities for members of a specified group.</li> <li>• Sets the LRU cache assignments to a device or device group.</li> <li>• Resets the LRU assignment back to the default LRU.</li> <li>• Lists the LRU cache assignments.</li> <li>• Enables the display of the SRDF director CPU workload distributions.</li> <li>• Creates and deletes cache partitions.</li> <li>• Modifies cache partitions.</li> <li>• Adds and removes a device or a range of devices to a cache partition.</li> <li>• Empties cache partitions.</li> <li>• Renames cache partitions.</li> <li>• Enables analysis of cache partitions.</li> <li>• Lists specific devices in a cache partition.</li> <li>• Enables/disables workload distribution on RDF directors.</li> <li>• Sets/resets workload distribution synchronous I/Os, asynchronous I/Os, and copy I/Os.</li> <li>• Lists the workload distribution settings for RDF directors.</li> </ul>

## Setting copy pace and priority

Using SYMCLI, you can set QoS parameters and values for specified devices managing the copy pace (priority) and the LRU assignments.

Also with Enginuity V5875, the pace and priority settings are supported for VLUNs and discontinued for BCV devices. In addition, the `symqos` command now supports storage groups as a target parameter.

## Copy pace and priority settings

Different copy pace (and priority) levels can be set on certain types of devices. Possible copy pace levels on the devices are 0, the highest, to 16, the lowest. Zero is the default. Use the following form when setting the copy pace.

```
symqos -g DgName [-bcv | -nobcv]
symqos [<-sg SgName | -devs <SymDevStart:SymDevEnd | SymDevName
        [, <SymDevStart:SymDevEnd | SymDevName> . . .]>>]

set <BCV|RDF|MIR|CLONE|VLUN> pace|priority
    <URGENT|Value|STOP> [LdevName]
```

If a device group is not specified, a storage group or devices must be specified. The definitions for the syntax follow.

Where:

- dg — Indicates the copy operation is on a device group devices.
- sg — Indicates the copy operation is on storage group devices.
- bcv — Specifies to only work on BCV group members.
- nobcv — Specifies to work on group members that are not BCVs.
- devs — Indicates the copy operation is on a range of devices.

BCV — TimeFinder copy operations.

---

**Note:** BCV is not supported for the URGENT and STOP values.

---

RDF — SRDF copy operations

MIR — Mirror copy operations

CLONE — Clone (previously called Snap) copy operations

VLUN — VLUN copy operations

---

**Note:** Setting the pace/priority for VLUN requires Enginuity 5875 and higher.

---

Unless an *LdevName* is specified, this command defaults to setting parameters on STD and BCV devices.

URGENT — The copy pace is classified as urgent and may be faster than the default.

*Value* — An integer value between 0 (fastest pace) to 16 (slowest pace). If the value URGENT is used, the copy pace is classified as urgent, and may be faster than 0, the default value. If the value STOP is used, copying is stopped.

---

**Note:** The URGENT and STOP actions are supported with Enginuity 5875 and higher.

---

stop — The background initiated copying will stop.

---

**Note:** The syntax items *pace* and *priority* are interchangeable nouns, which can be used to identify and set the same parameter.

---

With Engenuity 5875 and higher, the copy is driven from the target DA. This change increases optimized writes, and limits cache consumption, allowing tighter control of copy scheduling. QoS can still be applied on the target or source volumes, unless there are multiple clone sessions on the source device. In that case, applying QoS on the source will apply to all copy sessions. Setting on the target will only apply to that particular session.

You can specify a different clone copy priority for each clone session associated with the same source device. To set a separate clone priority for each session, the `set` operation needs to be performed on the clone target device. When the clone priority is set on the clone source device, all clone sessions associated with that device have the same clone priority. When priority is set on both the source and target devices, the target priority takes precedence and the source clone priority for that session is ignored.

### Examples

The `ProdA` group has 1000 volumes and is in the Precopied state, and the `ProdB` group has 10 volume pairs, all of which are also source devices for `ProdA`.

If devices in `ProdB` require copying quickly, set QoS only on the target devices to URGENT. Or alternatively, set QoS on `ProdA` to priority 16 to slow the copy of the 1000 volumes.

For SRDF operations, in device group `Proddb`, to set the copy pace level to 2 for all STD and BCV devices, enter:

```
symqos set RDF pace 2 -g Proddb
```

For SRDF operations, to set the SRDF copy pace value to 1 on just clone devices of device group `Proddb`, enter:

```
symqos set CLONE pace 1 -g Proddb -bcv
```

To set a copy pace to 1 during copy operations for mirror device `DEV021` in group `Proddb`, enter:

```
symqos set MIR pace 1 -g Proddb DEV021
```

## Using query

The following command and sample output shows the display for the `symqos query` command for a device group:

```
symqos -g dg_test1 query
```

```
Disk Group (DG) Name      : dg_test1
DG's Type                 : REGULAR
DG's Symmetrix ID        : 000194900341
```

Device Name				Copy Pace				
Logical	Physical	Sym	Config	BCV	RDF	MIR	CLN	VLN
DEV001	Not Visible	0170	Unprotected	0	2	0	0	URG
DEV002	Not Visible	0171	Unprotected	0	2	0	STP	0
. . .								

Legend:

```
Copy Pace : STP = Copy is stopped,
            URG = Copy has urgent priority.
```

## Listing copy pace settings

You can list copy pace settings for devices, a device group, or a storage group.

When copy pace is set on a storage group, the output only shows data specific to that storage group, as shown in the following example command and output:

```
symqos list -sid 432 -sg sg_test
```

```
Storage Group (SG) Name: sg_test
SG's Symmetrix ID      : 000194900432
```

Device Name		Copy Pace					
Sym	Physical	Config	BCV	RDF	MIR	CLN	VLN
0029	Not Visible	2-Way Mir	0	2	0	0	0

When copy pace is set on a devices, the Symmetrix ID displays at the top, as shown in the following example command and output:

```
symqos list -sid 311 -devs 0040:0048
```

```
Symmetrix ID      : 000194900311
```

Device Name		Copy Pace					
Sym	Physical	Config	BCV	RDF	MIR	CLN	VLN
040	Not Visible	2-Way Mir	0	3	0	0	0
041	Not Visible	2-Way Mir	0	3	0	0	0
042	Not Visible	2-Way Mir	0	3	0	0	0
043	Not Visible	2-Way Mir	0	3	0	0	0
044	Not Visible	2-Way Mir	0	3	0	0	0
045	Not Visible	2-Way Mir	0	3	0	0	0
046	Not Visible	2-Way Mir	0	3	0	0	0
047	Not Visible	2-Way Mir	0	3	0	0	0
048	Not Visible	2-Way Mir	0	3	0	0	0

When copy pace is set on a device group, the output only shows data specific to that device group, as shown in the following example command and output:

```
symqos -g dg_test list
```

```
Device Group (DG) Name: dg_test
DG's Type      : REGULAR
DG's Symmetrix ID : 000194900311
```

Device Name		Copy Pace					
Sym	Physical	Config	BCV	RDF	MIR	CLN	VLN
040	Not Visible	2-Way Mir	0	3	0	0	0
041	Not Visible	2-Way Mir	0	3	0	0	0
042	Not Visible	2-Way Mir	0	3	0	0	0
043	Not Visible	2-Way Mir	0	3	0	0	0
044	Not Visible	2-Way Mir	0	3	0	0	0
045	Not Visible	2-Way Mir	0	3	0	0	0
046	Not Visible	2-Way Mir	0	3	0	0	0
047	Not Visible	2-Way Mir	0	3	0	0	0
048	Not Visible	2-Way Mir	0	3	0	0	0

## Setting host I/O priority

Using Symmetrix Priority Controls (SPC), you can use the `symqos` command to enable and prioritize the service time of the host I/O to a device, a range of devices, or devices in a device or storage group. The following device types can not be assigned a priority:

- ◆ DATA
- ◆ DLDEV
- ◆ DRV
- ◆ SAVE
- ◆ SFS
- ◆ Thin
- ◆ VAULT
- ◆ VDEV

### Enabling and disabling priority of service

To enable or disable the priority of service feature for an array, use the following form:

```
symqos -pst -sid SymmID enable | disable
```

For example, to enable host I/O device priority feature for Symmetrix 1234, enter:

```
symqos -pst -sid 1234 enable
```

To disable host I/O device priority feature for Symmetrix 1234, enter:

```
symqos -pst -sid 1234 disable
```

**Note:** Enabling priority of service is not supported on Symmetrix VMAX 10K/VMAXe arrays.

### Setting host I/O priority

When priority of service is enabled, all eligible devices in the array are assigned priority 1 by default. To assign a new priority, use the following form:

```
symqos -pst -sid SymmID
[-devs <SymDevStart:SymDevEnd | SymDevName
      [, <SymDevStart:SymDevEnd | SymDevName>...]>] |
-g DgName] [-std] [-bcv]

set hostio priority Value
```

Where:

`-pst` — Indicates a priority service time operation.

`set hostio priority` — Assigns host I/O priority to a range of devices or devices in a device group. When setting the priority for devices in a device group, the priority can optionally be assigned to STD devices, BCV devices, or both.

*Value* — For host I/O priority setting, values are range from 1 (highest priority) to 16 (lowest priority).

For example, to assign all STD devices of device group `DeviceGroup` to host I/O priority 4, enter:

```
symqos -pst -g DeviceGroup set hostio priority 4 -std
```

If neither STD or BCV is specified, the default is STD.

## Listing priority of service

To view the priority of service for a range of devices or devices in a device group, or a given host I/O priority, use the following form:

```
symqos -pst

list -sid SymmID [-devs <SymDevStart:SymDevEnd | SymDevName
    [, <SymDevStart:SymDevEnd | SymDevName>...]>]
    [-all | -hostio_priority <<Value|ALL>]

list -g DgName [-std] [-bcv]
    [-hostio_priority <Value|ALL>]
```

For example, to list the priority of service time for Symmetrix 131, devices 002 and 003, enter:

```
symqos -pst -sid 131 list -hostio_priority -devs 002:003
```

The following is sample output from this command:

```
Symmetrix ID          : 000190300131
Priority Service Time State : Enabled

      Device          Priority
-----
Sym   Config          Attr HostIO
----
0002  2-Way Mir        -      2
0003  2-Way Mir        (M)   2
```

When listing device groups, if the device type is not specified, the display will list both STD and BCV devices.

For complete syntax for the `symqos` command, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

## Setting and resetting LRU parameters

Using SYMCLI, you can set QoS parameters and values for specified devices managing LRU cache I/O operations.

The contention for cache access can be managed by the least recently used (LRU) ring partitions in the Symmetrix cache. You can manage the cache strategy utilizing LRU rings (groups) and assigning certain devices or device groups to them. The syntax follows:

```
set LRU <#, #, # | ALL> [LdevName]
set LRUname LRUname [LdevName]

reset LRU [LdevName]
```

To assign all devices of device group `ProdDB` to LRU group 2, enter:

```
symqos -g ProdDB set LRU 2
```

Or, to assign all devices of device group `ProdDB` to an LRU group named `GROUP_2`, enter:

```
symqos -g ProdDB set LRUname GROUP_2
```

Using the `-lru` option of `symcfg list`, you can list the cache-slot allocation and allocation percentage of a specified LRU cache management group number. You can list all the LRUs if you specify `ALL`.

Using the `reset` option of `symqos`, you can reset the LRU assignment back to the default LRU. For example:

```
symqos reset LRU LdevName
```

## Setting RDF director workload distribution

Solutions Enabler V7.4 supports mixed SRDF workloads on the same RDF director. The supported workloads for this feature are:

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

The configuration can be done on a source or target Symmetrix array, but it only affects the director that contains the R1 devices in the RDF group. You may want to configure both sides the same if your devices ever swap the RDF personality.

Solution Enabler V7.4 provides an interface to enable and disable the RDF director CPU resource distribution and to display the current setting. If the feature is disabled, then the pre-5876 Enginuity behavior for RDF director CPU resource distribution will be in effect.

Use the `symqos` command to set and display the default RDF director CPU resource distribution assigned to each type of workload. The default RDF director CPU distribution will be used for an RDF director if the feature is enabled and an RDF director specific value has not been set or has been reset.

Use the `symqos` command to set, reset, and display the RDF director CPU resource distribution assigned to each type of workload on a specific RDF director.

---

**Note:** Any change made to a specific SRDF director's settings will take effect immediately for that director. Any change made to the system default settings will take effect immediately for all SRDF directors that do not have explicit director level settings.

---

### Feature requirements

This feature requires the following:

- ◆ The Symmetrix array must be running Enginuity 5876.
- ◆ The user must have `SYMAPI_ACCESS_TYPE_CFGSYM` access control privilege to the Symmetrix array involved in the action.
- ◆ The Symmetrix array must have an SRDF license.



## Enabling and disabling the RDF director workload distribution

Use the following syntax to enable this feature:

```
symqos -RA -sid SymmID

    enable -io
    disable -io
```

*Example* To enable the workload percentage settings for Synchronous, Asynchronous, and Copy I/Os on Symmetrix 1234, enter:

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

## Setting the RDF director workload distribution

Use the following syntax to set the I/O defaults on the SRDF director:

```
symqos -RA -sid SymmID

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

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

Each percent value is the percentage of RDF director CPU resources for each type of I/Os. These specified percentages should add up to 100.

*Examples* To set the RDF director workload percentages on Symmetrix 1234 to 60% for Synchronous I/Os, 30% for Asynchronous I/Os and 10% for Copy I/Os, enter:

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

To set the RDF 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, enter:

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

## Resetting the RDF director workload distribution

Use the `reset` command to reset the workload percentages to the Symmetrix array default values:

```
reset IO -dir <# | ALL>
```

*Example* To reset the settings of the workload percentages on director 8G of Symmetrix 1234, enter:

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

## Displaying the RDF director workload distribution

The following example shows the output format of the `symqos -RA list -io` command:

```
symqos -RA list -io

Symmetrix ID: 000194900300

RA IO State      : Disabled

System Defaults:
```

```

Synchronous IOs (%) : 50
Asynchronous IOs (%) : 40
Copies IOs (%)      : 10

```

RDF Directors:

	Flg	IO Percent		
Dir	R	Sync	Async	Copy
---	---	----	-----	----
08G	X	60	30	10
07H	.	50	40	10
08H	X	50	30	20

Symmetrix ID: 000194900397

RA IO State : Enabled

System Defaults:

```

Synchronous IOs (%) : 70
Asynchronous IOs (%) : 20
Copies IOs (%)      : 10

```

RDF Directors:

	Flg	IO Percent		
Dir	R	Sync	Async	Copy
---	---	----	-----	----
08G	X	60	30	10
07H	.	70	20	10
08H	X	50	30	20

Symmetrix ID: 000194901138

RA IO State : N/A

System Defaults:

```

Synchronous IOs (%) : N/A
Asynchronous IOs (%) : N/A
Copies IOs (%)      : N/A

```

RDF Directors:

	Flg	IO Percent		
Dir	R	Sync	Async	Copy
---	---	----	-----	----
07G	-	-	-	-
08G	-	-	-	-
07H	-	-	-	-
08H	-	-	-	-

Legend for Flg:

(R)A IO Set: X = Set, . = Default, - = N/A

The command in this example is run without the Symmetrix ID (-sid), therefore the information for all Symmetrix arrays and all RDF directors displays.

A Symmetrix array running an Engenuity version lower than 5876 displays a dash (-) and N/A, as shown for Symmetrix 138.

## Viewing the audit log

The sample audit log entries shown next are the generated logs for the following `symqos` commands respectively:

```
symqos enable -IO
symqos -RA -sid 397 set IO -default -sync 60 -async 30 -copy 10
symqos -RA -sid 397 set IO -dir 8G -sync 70 -async 10 -copy 20

symaudit list -text -sid 397
```

```

      A U D I T   L O G   D A T A

Symmetrix ID           : 000194900397

Record
Number   Date       Time       Function  Action  Activity
-----
Text
-----
1541  11/14/11  16:02:08  QoS      Enable  SEbe71a374621
      STARTING a QOS 'ENABLE_RA_IO' operation. Symm= 000194900397

1542  11/14/11  16:03:42  QoS      Enable  SEbe71a374621
      The QOS 'ENABLE_RA_IO' operation SUCCESSFULLY COMPLETED

. . .

1788  11/14/11  16:06:13  QoS      Set     SEbe71a99355
      STARTING a QOS 'SET' operation. Symm= 000194900397, SyncPercent=60,
      AsyncPercent=30, CopyPercent=10

1789  11/14/11  16:07:38  QoS      Set     SEbe71a99355
      The QOS 'SET' operation SUCCESSFULLY COMPLETED

. . .

```

## Managing dynamic cache partitions

Dynamic Cache Partitioning (DCP) divides the cache memory into multiple partitions with unique names and their device path assignments. Partition areas can be made static or dynamic in size. The dynamic partitioning provides flexibility to the amount of floating memory that can be allocated with a high and low watermark. This allows memory resources to be temporarily donated to other partitions when needed. The `symqos` command allows you to create partitions for different device groupings in addition to the default partition that all devices belong to initially. Each partition has a target cache percentage as well as a minimum and maximum percentage. In addition, you can donate unused cache to other partitions after a specified donation time. With Solutions Enabler V7.4 and higher, the number of allowed cache partitions is 16.

Cache partitioning is enabled and disabled on the Symmetrix array.

### Prerequisites

Before creating a cache partition, note the following:

- ◆ There must be an available partition.

- ◆ There must be enough cache left in the default partition that it does not fall below the minimum required cache.

## Enabling and disabling cache partitions

To enable or disable cache partitions for the Symmetrix array, use the following form:

```
symqos -cp -sid SymmID enable | disable
```

### Examples

To enable cache partition settings on Symmetrix 301, enter:

```
symqos -cp -sid 301 enable
```

To disable cache partition settings on Symmetrix 301, enter:

```
symqos -cp -sid 301 disable
```

## Creating cache partitions

Before you can add devices to a cache partition, it must be created. To create a cache partition, use the following form:

```
symqos -cp -name PartitionName -sid SymmID

create -target TargetPercent
      -min MinimumPercent
      -max MaximumPercent
      -wp WritePendingLimit
      -time DonationTime
```

Where:

**-cp** — Indicates a cache partition operation.

*PartitionName* — Identifies the partition by a unique name. The name can be up to 31 characters.

*TargetPercent* — Specifies the target cache percentage for a cache partition.

*MinimumPercent* — Specifies the minimum cache percentage for a cache partition.

*MaximumPercent* — Specifies the maximum cache percentage for a cache partition.

*WritePendingLimit* — Specifies the write pending limit percentage for a cache partition. Possible values are 40 - 80.

*DonationTime* — Specifies the donation time in seconds; the time for when idle cache is made available to other cache partitions.

### Example

To create a cache partition named `TestPartition` on Symmetrix 1234, enter:

```
symqos -cp -name TestPartition -sid 1234 -target 10 -min 5 -max 40 -wp 50 -time 10 create
```

## Adding devices to a cache partition

You can add single devices, a range of devices, or devices in a device group or SRDF group to a cache partition. The `addall` option must be used with device ranges, SRDF groups, or pools. With device groups, you can specify all devices (`addall`), logical devices (`ld`), standard devices (`-std`), BCV devices (`-bcv`), or virtual devices (`-vdev`).

## Device restrictions

When adding devices to a cache partition:

- ◆ Only specify the head device for metadevices and CKD devices.
- ◆ Do not add VAULT devices, as these are blocked from being moved to any new partition.
- ◆ Ensure that all devices in SRDF groups that are in Async mode belong to a single partition.
- ◆ With Enginuity version 5876 and higher, if a device range is specified with the `addall` or `rma11` commands, DATA devices are included in the operation. In previous versions of Enginuity, DATA devices were blocked.

To add devices to a partition, use the following form:

```
symqos -cp -name PartitionName -sid SymmID
    add dev SymDevName

symqos -cp -name PartitionName -sid SymmID
    [-devs <SymDevStart:SymDevEnd | SymDevName
        [, <SymDevStart:SymDevEnd | SymDevName>...]>] |
        -rdfg GrpNum | -pool PoolName -snap | -rdfa_dse | -thin]
    addall

symqos -cp -name PartitionName -g DgName
    add ld LdevName
    addall [-std] [-bcv] [-vdev]
```

If the device type is not specified, the default is to add only STD devices.

#### Examples

To add device 001 to partition `TestP` on Symmetrix 310, enter:

```
symqos -cp -name TestP -sid 310 add dev 001
```

To add device range 002 through 012 to partition `TestP` on Symmetrix 310, enter:

```
symqos -cp -name TestP -sid 310 -devs 002:012 addall
```

To add the BCV devices in device group `Finance` to the partition `TestP` on Symmetrix 310, enter:

```
-symqos -cp -name TestP -sid 310 -g Finance addall -bcv
```

With Enginuity 5876 and higher, device pools can be added to and removed from cache partitions.

To add a thin pool named `Finance` to the cache partition `TestA`, enter:

```
symqos -cp TestA -sid 310 addall -pool Finance -thin
```

## Modifying a cache partition

Use the `modify` argument to change the values of a cache partition, as follows:

```
symqos -cp -name PartitionName -ran-sid SymmID

    modify [-target TargetPercent]
           [-min MinimumPercent]
           [-max MaximumPercent]
           [-wp WritePendingLimit]
           [-time DonationTime]
```

The parameter definitions are listed earlier in [“Creating cache partitions” on page 276](#).

## Renaming a cache partition

Use the `rename -new_name` option to rename the cache partition, as follows:

```
symqos -cp -name PartitionName -sid SymmID
      rename -new_name NewPartitionName
```

For example, to rename cache partition `TestP` on Symmetrix 310 to `TestA`, enter:

```
symqos -cp TestP -sid 310 rename -new_name TestA
```

## Deleting a cache partition

Use the `symqos delete` command to delete a cache partition. When deleting a cache partition, all devices, groups, and pools currently assigned to that partition will move back to the default partition.

## Removing devices from a cache partition

Use the `remove dev`, `rmall`, and `remove ld` commands to remove devices from a cache partition and return them to the default partition. This can be done for a device range, a device group, SRDF group, or pool.

To remove the BCV devices in device group `Finance` from the partition `TestP` on Symmetrix 310, enter:

```
symqos -cp -name TestP -sid 310 -g Finance rmall -bcv
```

## Using analyze mode

Dynamic Cache Partitioning (DCP) provides an `analyze` mode, which you can use to determine how much actual cache your applications require. While in `analyze` mode, applications will not fail due to cache partition overflow.

Cache partitions cannot be changed from enabled to `analyze`; the cache partition must be disabled to use the `analyze` mode. The syntax for setting the cache partition to `analyze` mode follows:

```
symqos -cp -sid SymmID
      analyze
```

To run in `analyze` mode, set the target percent, and then set all partitions with a minimum of 0%, a maximum of 100%, and a donation time of 0 seconds. After running in `analyze` mode, you can display the cache partition settings with the `-usage` option, and determine more accurate cache partition target values.

The `-usage` option is shown in the following sample command and output:

```
symqos -cp list -sid 237 -settings -usage
```

```
Symmetrix ID: 000190300237
```

```
Cache Partitioning      : Enabled
Number of Partitions    : 2
Max Num of Partitions   : 8
Min Allowed Target %    : 10
Max Allowed Target %    : 90
XRC Partition State     : Enabled
XRC Partition Name      : XRC_partition
Empty Partition Removal : Remove
Time of Last Modification : 01:12:02 PM on Tue Mar 13,2007
```

Partition Name	Min (%)	Tgt (%)	Max (%)	WP (%)	Time (s)	Device Count	Cache Slots		Used (%)
							WP	Used	
DEFAULT_PARTITION	0	65	100	80	300	1024	2156	395166	95
Lab2	35	35	35	50	300	56	234	6898	65

## SRDF/A output

To have the SRDF/A output based on cache partition values, instead of system-wide settings, set `SYMAPI_CP_ANALYSIS_MODE` in the options file to `ENABLE` when running in `analyze` mode.

The cache partition values display automatically when cache partitions are enabled, but only display in `analyze` mode when this options file setting is enabled. The default value is `DISABLE`.

The SRDF/A information displayed in CLI commands will show the percent of cache in use, calculated by the target cache percentage and the write pending limit for the partition of the associated SRDF group.

```
RDFA Information:
{
    Session Number           : 39
    Cycle Number             : 0
    Number of Devices in the Session : 16
    Session Status           : Inactive
. . .
    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
. . .
```

## Empty partition removal

By default, empty partitions or partitions without assigned devices are removed after two hours with no activity. This feature appears in the partition settings output (see the DCP settings output).

To change this feature, use the following form:

```
symqos -cp -sid SymmID set empty preserve
```

## Listing cache partitions

When listing cache partitions, the information can be for a range of devices, a device group, an SRDF group, a Symmetrix array, or a cache partition name. List cache partition information using the following form:

```
symqos -cp [-name PartitionName]

list [-devs SymDevStart:SymDevEnd [-all]] [-rdfg GrpNum]
list -g DgName [-std] [-bcv] [-vdev]
list [-sid SymmID] -settings [-usage] [-v]
```

The `-settings` option lists the general cache partition configuration for the array, as shown in the following example:

```
symqos -cp -sid 237 list -settings
```

```
Symmetrix ID: 000190300237
```

```
Cache Partitioning      :Analysis
Number of Partitions    :2
Max Num of Partitions   :8
Min Allowed Target %    :10
Max Allowed Target %    :90
XRC Partition State     :Enabled
XRC Partition Name      :XRC_partition
Empty Partition Removal :Enabled
Time of Last Modification :2:11:47 PM on Thu May 24, 2007
```

Partition Name	Min (%)	Tgt (%)	Max (%)	WP (%)	Time (s)
-----	---	---	---	---	-----
DEFAULT_PARTITION	0	100	100	80	300

The `-usage` option includes cache usage information, as shown in the following example:

```
symqos -cp list -sid 237 -settings -usage
```

Symmetrix ID: 000190300237

```
Cache Partitioning      : Enabled
Number of Partitions    : 2
Max Num of Partitions   : 8
Min Allowed Target %    : 10
Max Allowed Target %    : 90
XRC Partition State     : Enabled
XRC Partition Name      : XRC_partition
Empty Partition Removal :Enabled
Time of Last Modification : 01:12:02 PM on Tue Mar 13,2007
```

Partition Name	Min (%)	Tgt (%)	Max (%)	WP (%)	Time (s)	Device Count	Cache Slots		Used (%)
							WP	Used	
-----	---	---	---	---	-----	-----	-----	-----	-----
DEFAULT_PARTITION	0	65	100	80	300	1024	2156	395166	95
Lab2	35	35	35	50	300	56	234	6898	65

For complete syntax of the `symqos` command, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

# QoS examples

Using SYMCLI, you can get a list of QoS parameters and values for specified devices.

For example, to get a list of copy pace levels for devices 010 through 0B2 in Symmetrix 2334, enter:

```
symqos list -sid 2334 -devs 10:B2
```

**Note:** If you omit the `-sid` option, the specified range of devices will apply to all Symmetrix arrays.

In the preceding example, concerning the `-devs` option, the *SymdevStart* value will default to zero if you omit the value as follows:

```
symqos list -sid 2334 -devs :B2
```

This will increase the range to include all devices between 000 and 0B2. If you omit the *SymdevEnd* value, the range will list all devices found beyond the *SymdevStart* value on the specified Symmetrix array as follows:



```
symqos list -sid 2334 -devs 10:
```

To list the LRU assignments of devices 00C through 013 to the various LRU cache management groups, enter:

```
symqos list -lru -sid 346 -devs 00C:013
```

To get the copy pace levels concerning just the BCV devices of device group `ProdDB`, enter:

```
symqos -g ProdDB query -bcv
```

To get the copy pace levels concerning both the BCV and STD devices of device group `ProdDB`, enter:

```
symqos -g ProdDB query
```

To get the copy pace levels concerning just the STD devices of device group `ProdDB`, enter:

```
symqos -g ProdDB query -nobcv
```

To view the LRU assignment of all devices in a device group, enter:

```
symqos -g ProdDB query -lru
```

To assign all devices of device group `ProdDB` to LRU group 2, enter:

```
symqos -g ProdDB set LRU
```

To enable the workload percentage settings for Synchronous, Asynchronous, and Copy I/Os on Symmetrix 1234, enter:

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

To set the default settings of the workload percentages on Symmetrix 1234 to 60% for Synchronous I/Os, 30% for Asynchronous I/Os, and 10% for Copy I/Os, enter:

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

To change 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, enter:

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

To reset the settings of the workload percentages on director 8G of Symmetrix 1234, enter:

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



# CHAPTER 9

## Optimizing Array Performance

This chapter describes the Symmetrix Optimizer concepts and how to improve array performance using the Symmetrix Optimizer commands of the SYMCLI. This chapter covers the following topics:

◆ What is Symmetrix Optimizer? .....	284
◆ How does Symmetrix Optimizer work? .....	287
◆ Symmetrix Optimizer operations using SYMCLI .....	293
◆ Setting control parameters .....	295
◆ Setting advanced parameters .....	295
◆ Setting time windows .....	296
◆ Setting the swap priority for devices .....	296
◆ Setting manual swap lists .....	297
◆ Reviewing Optimizer plans .....	297
◆ Approving a swap list .....	306
◆ Performing rollbacks .....	307
◆ Reading Optimizer logs .....	307
◆ Migrating devices .....	308

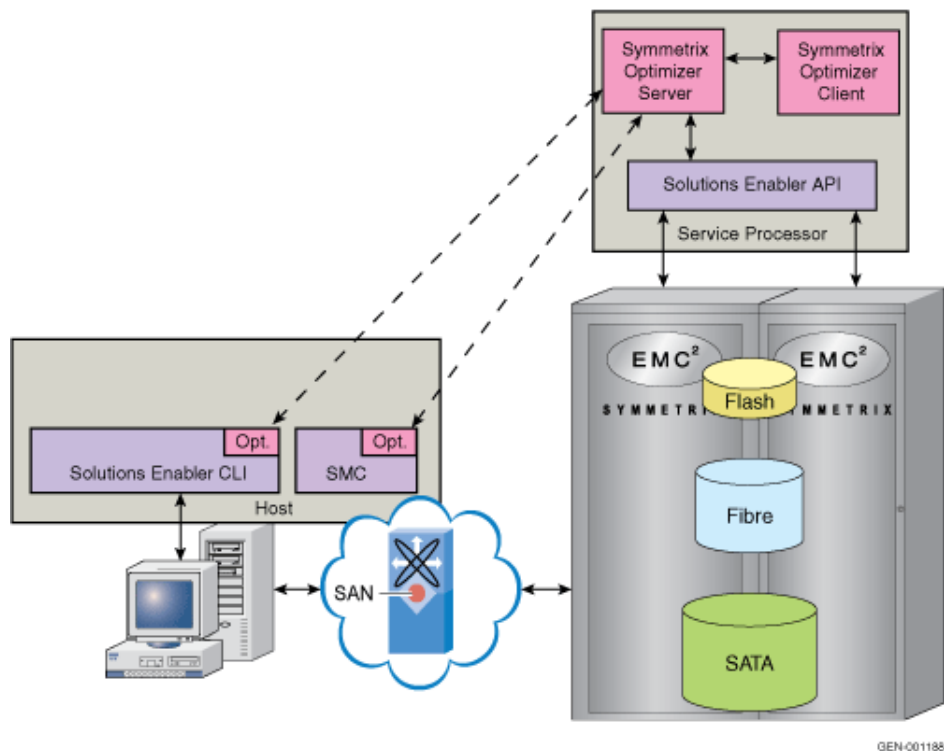
## What is Symmetrix Optimizer?

Symmetrix Optimizer is one of a family of information management software solutions offered by EMC. Symmetrix Optimizer improves array performance by continuously monitoring the back-end activity and swapping highly active volumes with idle volumes to achieve improve the balance of the workload across the disks. This automated, continuous process is based on user-defined parameters and is completely transparent to end users, hosts, and applications in the environment.

Solutions Enabler V7.1 includes a significant performance improvement for Symmetrix Optimizer software by changing the storage location of the Optimizer control settings—from the Optimizer server to the Symmetrix array. In addition, locks are no longer taken on the Optimizer server. Because of these changes, the Optimizer server will not have to stop the background statistic collection and swap processes when you set parameters and define swaps.

**Note:** Solutions Enabler V7.1 introduces Fully Automated Storage Tiering (FAST), which provides the ability to move volumes between multiple storage tiers (Flash, Fibre Channel, SATA). This first release of FAST shares the Optimizer database on the Symmetrix service processor—and uses the same SYMCLI control parameters and time windows. [Chapter 4, “Fully Automated Storage Tiering,”](#) explains how to use FAST.

The Symmetrix Optimizer application (client and server) resides on the Symmetrix service processor, as shown in [Figure 24](#).



**Figure 24** Symmetrix Optimizer application architecture

The Optimizer client interface on the service processor is used by EMC Customer Service. From a host system, Solutions Enabler CLI commands and Symmetrix Management Console (SMC) dialogs can communicate with the Optimizer server.

The SYMCLI command, `symoptmz` provides all the necessary commands to control the behavior of Symmetrix Optimizer.

This chapter provides an overview of the Optimizer functionality, its requirements and restrictions, the different modes of operation, and a description of the process Optimizer uses to make the device swaps.

The second half of the chapter introduces the `symoptmz` command, and explains all the options, variables, and parameters that can be used to manage Symmetrix Optimizer on the Symmetrix array.

Symmetrix Optimizer always works within a single Symmetrix array.

Optimizer does not change the protection of a device.

## Data movement

Optimizer provides two types of data movement: full device swap and device move. In a device move operation, a set of source devices will be moved to a list of disks; the array internally determines the device distribution on the disks.

## Configuration requirements

Symmetrix Optimizer has evolved over time and configuration requirements have changed with Enginuity versions. This section provides the common requirements, then divides the requirements by Enginuity version.

## Devices

The following restrictions apply to devices that are considered for a swap:

- Same size and emulation
- Same protection (Enginuity 5874)
- Available DRVs for both source and target
- Currently attached to legal RAID groups

Symmetrix Optimizer can optimize all open system and mainframe RAID 1 (mirrored), RAID 5, RAID 6, and RDF-protected devices in an array. Optimizer collects statistics for all devices in the array, however, the devices listed below are not supported:

- ◆ BCV devices
- ◆ CKD meta
- ◆ DATA devices
- ◆ DRV devices
- ◆ Marked by user as "Do not move"
- ◆ SFS devices
- ◆ Striped CKD devices
- ◆ Thin device

- ◆ Vault devices
- ◆ Virtual / SAVE devices
- ◆ WORM protected devices

This means that statistics are still collected for them, but the devices will not be swapped by Optimizer.

## Enginuity 5773 and earlier

### DRV devices

As a minimum, two DRVs must be configured for each size and emulation of the volume to be swapped by Optimizer. For example, for Optimizer to swap a 4 GB open system volume, then two 4 GB open system DRVs must be configured. If more than one simultaneous swap is desired, then additional DRVs are needed (two for each swap). If there are volumes of mixed size or emulation within the same array, then DRVs are required for each size and type to be swapped.

### Open mirror position

Optimizer requires an open mirror position to be able to perform a swap, however, Symmetrix arrays (running Enginuity 5773 and earlier) have a four mirror slot limitation per volume. This limitation may make it difficult in certain configurations for Optimizer to obtain the mirror position it requires to perform a swap. For example, a RAID 1 device already has two BCVs established, and Optimizer wants to swap this device, it cannot get a mirror slot for the DRV. If Optimizer does not have a mirror slot available, then it will drop the swap and rerun the analysis. If Optimizer is in Rollback, Manual, or User-Approved mode, then Optimizer will retry a number of times until an error is encountered. Eventually, Optimizer will stop trying if a mirror slot does not become available.

### Configuration lock requirement

For Symmetrix arrays running 5773 and earlier, Optimizer holds a Symmetrix External Lock (SEL) to perform a swap or migration. The configuration lock is an exclusive lock on the Symmetrix array for performing configuration changes. This lock prevents multiple applications from changing the array configuration at the same time. If another application holds the configuration lock when Optimizer wants to create a swap, then Optimizer will behave as previously described (no open mirror position). Optimizer will hold the configuration lock for the duration of the swap or migration, or until the operation is cancelled, so other applications will not have the ability to perform configuration changes during this time.

---

**Note:** Committed swap or migration operations that are scheduled for a later time can be cancelled up until the scheduled time.

---

## Enginuity 5874 and higher

### DRV devices

All DRVs are mirrored and only one DRV per swap is required. In addition, the DRV can be equal or larger in size than the logical device that is swapped.

## Open mirror position

Symmetrix arrays running Enginuity 5874 and higher do not require the open mirror position.

## Open configuration lock requirement

Swaps on Symmetrix arrays running Enginuity 5874 and higher do not explicitly prevent other configuration changes from taking place during the swap operation.

# How does Symmetrix Optimizer work?

The analysis methodology and the swap procedure are the core technology and the two main components of the Symmetrix Optimizer solution. The following sections provide a brief description of each.

Once initialized with the user-defined parameters, Symmetrix Optimizer operates totally autonomously on the Symmetrix service processor. Symmetrix Optimizer monitors and controls array performance by following these steps:

**Step 1:** Symmetrix Optimizer builds a database of device activity statistics on the Symmetrix back-end. This data is saved for a maximum of two weeks.

**Step 2:** Using the statistical data collected, configuration information, and the user-defined parameters, the Optimizer algorithm identifies busy and idle devices and their locations on the physical drives. The algorithm tries to minimize average disk service time by balancing I/O activity across physical disks. Optimizer determines which disks require balancing by locating busy devices close to each other on the same disk, and/or by locating busy devices on faster areas of the disks. Optimizer takes into account the speed of the disk, the disk geometry, and the actuator speed to determine faster disks.

**Step 3:** Once a solution for load balancing has been developed, the next phase is to carry out the Symmetrix device swaps. You can specify whether swaps should occur in a completely automated fashion, or if the device swaps require user approval before the action is taken.

**Step 4:** Once a swap function completes, Symmetrix Optimizer continues data analysis for the next swap.

## Performance metrics

Optimizer looks only at back-end activity; it uses the back-end logical device statistics, and the following metrics:

- ◆ DA logical volumes reads
- ◆ DA logical volumes writes
- ◆ Logical volume prefetch
- ◆ DA logical volumes KB read
- ◆ DA logical volumes KB written

While modeling the disk service time, different weights are assigned to read, write, and prefetch activity. The assumption is that since writes are done as a background process, they are done in sequence and hence are more efficient. Optimizer does not try to follow DMSP policies; instead, it assumes that reads are equally spread among all the device mirrors, and all mirrors perform all writes.

## Modeling service time

Service time is defined as the sum of seeks, latency and transfer time:

- ◆ **Seek time** — Seek time is the time it takes the disk arm to move and position the disk head on the correct track—move from track *X* (serving the previous I/O) to track *Y* (serving the next I/O). Optimizer uses a gig-to-gig database to model seek time for different addresses and different disk drives.
- ◆ **Latency time** — Latency time is the delay for disk rotation. The latency time is a function of the disk rotational speed. Optimizer assumes one-third of a spin for each I/O, assuming the internal disk optimization is on.
- ◆ **Transfer time** — Transfer time is the time that it takes the disk to transfer the data from/to the disk. Transfer time is a function of the data transfer rate, disk bandwidth, and data layout. Optimizer uses the Zone Bit Recording (ZBR) database to model the transfer time of data. The ZBR database includes information about the bandwidth of each zone of the disk.

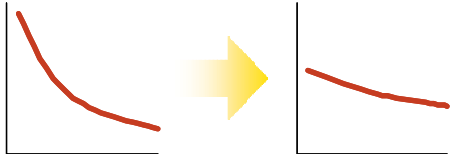
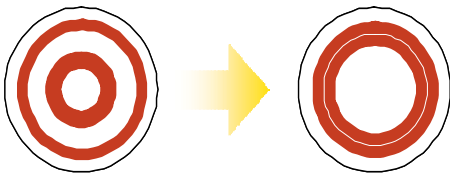
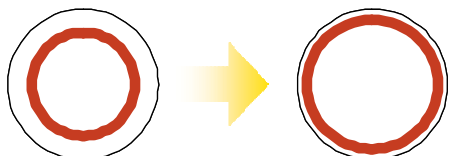
Accurate seek and latency times are impractical to get because they require a complete trace of I/Os sequence, therefore the Optimizer uses a mathematical model to calculate these metrics.



## The analysis

When Symmetrix Optimizer evaluates the performance statistics, it determines potential device swaps based on how well they would improve overall performance. This analysis is based on minimizing disk service time (rotational latency plus seek time plus transfer time). Symmetrix Optimizer uses three strategies when determining which swaps to make.

**Table 22** Optimizer strategies for determining swaps

Swap strategy	Description
<p><b>1. Load Balancing</b></p> 	Swaps highly active devices on disks with lower activity to even out the load across the physical drives in the Symmetrix system. This decreases the contention on the individual physical disks, improving the performance of both highly active devices and low-activity devices.
<p><b>2. Minimize Seek</b></p> 	Performs swaps that relocate highly active devices so that they are closer together on the physical disk and will decrease the seek distance for I/Os to these devices. When Optimizer swaps devices, it will try to achieve this type of configuration since it decreases the overall I/O service time.
<p><b>3. Use Faster Media</b></p> 	Optimizer tries to swap highly active devices on the outer zones of the disk. This is because devices located on the outer zones of the disk have faster transfer speeds.

Symmetrix Optimizer's algorithm uses detailed disk performance information that takes into account several drive characteristics, such as those gig-to-gig seek times, zone-bit recording, and bandwidth data.

## Finding the best swap

The Optimizer analysis consists of three high-level phases:

- ◆ **Calculate service time** — Model and sum the total service time of each disk and for every time stamp that was marked to be included by the analysis.
- ◆ **Sort disks by activity** — Sort all disks by their modeled total service time.
- ◆ **Find best swap** — Starting at the busiest disk, check all potential swaps. The analysis process models *what-if* scenarios using *virtual swaps* to estimate the impact of a swap on the service time of the affected disks. The philosophy of Optimizer is to check as many swaps as possible in order to guarantee that the best swap is indeed selected.

For Engenuity 5773 and earlier, 1 to 256 hypers are allowed per disk. The maximum number of hypers per physical disk automatically sets to 512 for Engenuity 5874 and 1024 for Engenuity 5875. Finding a swap that improves service time by 10 percent or more on these systems is very rare. Since using a percentage of improvement as swap *goodness* criteria is not applicable anymore, Optimizer had to adapt a different method (*MO*) that

deals better with the “many hypers contribute smaller chunks” problem. M0 is defined as “the best you could possibly get” from a swap. Each disk in the system is assigned an M0 designation, which is defined as the minimum service time a disk can get by replacing one of its hypers by a null hyper (a hyper that performs no I/Os to the disk).

When analyzing a swap, Optimizer checks how close the new-modeled service time is to the disk’s M0; the closer to M0, the better the swap. Usually, two disks are affected by a single swap; the busier disk’s service time is expected to go down, while the other disk’s service time is expected to go up. In addition to the M0, the Optimizer also ensures that the new maximum of service times is less than the old one.

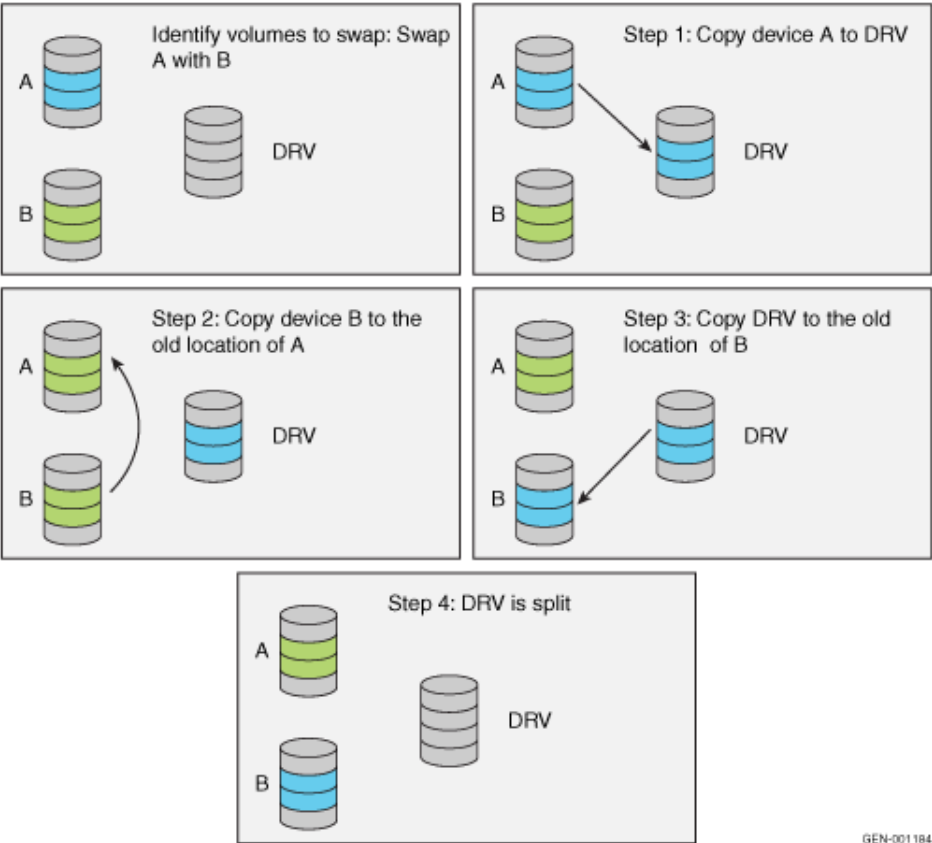
## Swap procedure for Enginuity 5874 and higher

Symmetrix VMAX Family arrays with Enginuity 5874 streamlines the device swap procedure by using only one DRV device during the swap. With Enginuity 5874, only mirrored DRV devices can be created. Refer to [Table 5 on page 43](#) for information about creating mirrored DRV devices.

This section provides the new steps for swapping devices using Symmetrix Optimizer. Refer to [“Swap procedure for Enginuity 5773 and earlier” on page 291](#) if you have an earlier version of Enginuity.

Hypervolumes are swapped using a four-step process. One DRV must be configured per swap. The DRV should be of equal or greater size than the swapped devices. With Symmetrix Optimizer, up to eight simultaneous swaps can happen at one time.

[Figure 25 on page 291](#) illustrates the steps for swapping device A with device B.



GEN-001184

Figure 25 Swap steps using one DRV

### Swap procedure for Engenuity 5773 and earlier

The swap procedure relies on the Symmetrix TimeFinder technology and uses DRVs as temporary mirrors. The following section describes the four swap steps when dealing with mirrored devices swaps. In the case of three mirror devices, the swap procedure does not require a DRV.

#### Step 1: Identify volumes to swap

Symmetrix Optimizer identifies a pair of hypervolumes to swap by recognizable patterns of hypervolume activity and criteria. In Figure 26, assume the red volumes (on A) have high activity and the blue volumes (on B) have low activity.

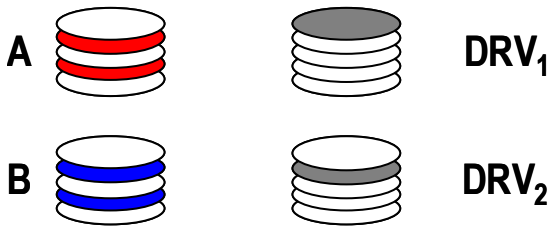


Figure 26 Identify volumes to swap

## Step 2: Copy volume to DRV

Symmetrix Optimizer swap commands are passed to SymmWin, which assigns one DRV (DRV1) as a third mirror for hypervolume A. A second DRV (DRV2) will be assigned as a third mirror for hypervolume B. All tracks on the third mirror are marked invalid. Tracks are copied from the valid mirrors to the two DRVs. After the DRVs are synchronized, the two original swap physical mirrors are marked Not Ready (volume A and volume B) and their attributes are swapped (see [Figure 27](#)). Both hypervolumes still have two (or more) physical mirrors. Host activity to the hypervolumes is now directed to DRVs and the other mirror.

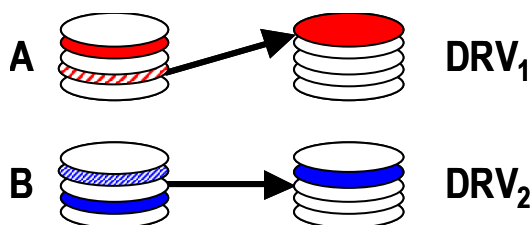


Figure 27 Copy volume to DRV

## Step 3: Copy DRV to new location

The data on the DRVs are now copied to the new location. After the attributes of the original hypervolumes are swapped, SymmWin copies the tracks from the valid mirrors to the two new mirrors and then makes the original hypervolumes Ready (see [Figure 28](#)). This is similar to a BCV restore.

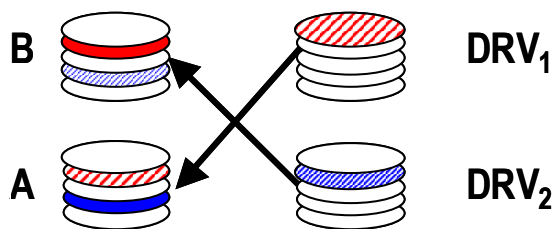


Figure 28 Copy DRVs to new locations

## Step 4: DRVs are split

The final step is to split the DRVs from their standard and mirror hypervolumes after synchronization completes (see [Figure 29](#)). The drive's balance improves and the DRVs are now available for the next swap.

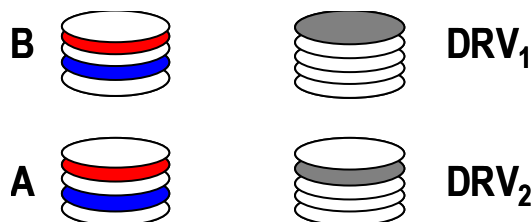


Figure 29 DRVs are split

**Note:** Swap of three mirror devices does not require a DRV.

## Legality of swaps

As a rule, Optimizer will never suggest a swap that conflicts with configuration rules as defined by the Engenuity, SymmWin, or the configuration groups. To minimize the affect of the swap procedure on the overall Symmetrix performance, the Optimizer will not include the same spindle twice in one swap group. In addition, you can use QoS controls to set a lower BCV priority for the STDs to be swapped, and plan the swap activities to occur on idle business time using the Time Window feature. [Chapter 8, “Managing Quality of Service”](#) explains how to use QoS controls.

## Symmetrix Optimizer operations using SYMCLI

SYMCLI on the host system issues commands to display the values of the parameters that control the behavior of the Symmetrix Optimizer. Some of the `symoptmz` commands are executed directly on the command line, while others are processed in a command file.

[Table 23](#) describes the `symoptmz` commands that can be executed on the command line.

**Table 23** Symmetrix Optimizer CLI command

<code>symoptmz</code> command argument	Description
<code>clear_stats</code>	Clears disk statistics maintained by Symmetrix Optimizer.
<code>commit</code>	Updates Optimizer with the changes defined in the command file.
<code>disable</code>	Disables the Optimizer algorithm processing.
<code>enable</code>	Enables the Optimizer algorithm processing.
<code>list</code>	Displays Optimizer-specific attributes of Symmetrix devices.
<code>prepare</code>	Performs extra range checks on the changes specified in the command file.
<code>preview</code>	Verifies the syntax of the changes specified in the command file.
<code>query</code>	Queries Symmetrix Optimizer and displays the current state and version information of the Optimizer. If <code>-v</code> is specified, additional information about the version and open sessions displays.
<code>read</code>	Reads from Optimizer’s log file.
<code>rollback</code>	Rolls back a Symmetrix configuration to that of a previous point-in-time.
<code>show</code>	Shows information about the current Optimizer data movement/swap plans or configuration parameters.
<code>sync</code>	Acquires version information from Optimizer. This argument is supported only in conjunction with <code>-version</code> .

This section describes the `symoptmz` commands listed in [Table 23](#). Refer to the following sections for command file commands:

[“Setting control parameters” on page 295](#)

[“Setting advanced parameters” on page 295](#)

[“Setting time windows” on page 296](#)

[“Setting the swap priority for devices” on page 296](#)

[“Setting manual swap lists” on page 297](#)

[“Reviewing Optimizer plans” on page 297](#)

[“Approving a swap list” on page 306](#)

[“Performing rollbacks” on page 307](#)

[“Reading Optimizer logs” on page 307](#)

[“Migrating devices” on page 308](#)

## Enabling and Disabling Optimizer

To start and stop the Optimizer process on the array service processor, use the `symoptmz enable` and `symoptmz disable` commands. When enabled, Optimizer collects data and performs data movement. When disabled, Optimizer does not perform data movement.

## Committing command file changes

---

**Note:** Solutions Enabler V7.1 has eliminated stopping Optimizer to process a request, such as setting parameters or defining a swap. The Optimizer start and stop state has been replaced by the `enable` and `disable` commands.

---

Commands for setting or clearing any parameters or time windows may be placed in a command file, which will then be processed by `symoptmz`. Alternatively, `stdin` redirection can be used with "here documents" in UNIX shell scripts.

Each command in the file has to be terminated by a semi-colon (;). There is no limit on the number of commands or the type of commands that can be placed in a command file. All the commands in a command file are executed in a single Optimizer session. The commands in the command file are not case sensitive, however, the parameters entered are case sensitive.

To change any Optimizer parameters, use a command file with the `commit` option.

### *Example*

To commit the changes specified in the command file `opt_config.txt`, issue the command:

```
symoptmz -file opt_config.txt commit
```

The `commit` option carries out syntax and range checks, and then updates the Optimizer with the modified parameters.

## Clearing the Optimizer statistics

To clear the disk statistics maintained by Optimizer, use the following form:

```
symoptmz -sid SymmID clear_stats
```

# Setting control parameters

**Note:** Control parameters can be set for Optimizer or FAST. Both tools use the same command file format.

Optimizer control parameters are set using the following syntax in the command file:

```
set control_parms [start_mode=AUTO | MANUAL>,]
[swap_mode=AUTO | USER_OK,]
[min_perf_period=min_perf,]
[workload_period=workload,]
[max_simult_swaps=max_simult,]
[swap_rate=max_swaps];
```

Table 24 describes each parameter.

**Table 24** Optimizer control parameters

Parameter	Description
start_mode	Determines whether Optimizer is enabled or disabled whenever Optimizer is launched. For example, after rebooting the service processor.
swap_mode	Controls whether Optimizer should automatically swap devices (AUTO) as soon as it finds swaps that would improve performance. If Optimizer is enabled in AUTO swap mode, each day it will make up to the number of swaps specified by the swap_rate parameter, under control of the swap time window settings. If Optimizer is enabled in USER_OK swap mode, it will generate lists of swap suggestions approximately once an hour, and then wait for the user to approve the swap list before proceeding.
min_perf_period	Specifies the amount of samples required initially before a recommendation will be made. You should make sure that the values you specify are long enough (usually a week) for Optimizer to establish a good characterization of your typical workloads. This parameter is expressed in hours. Keep in mind that the Optimizer statistics database holds about 14 days worth of data.
workload_period	Specifies how far back in time Optimizer should consider when the optimization algorithm is run. Be careful not to make this value too large or you may include data that is so old it is no longer representative of your current workloads. This parameter is expressed in hours.
max_simult_swaps	Controls how many swaps Optimizer can perform simultaneously (up to four). The actual value should reflect the number of devices that will be swapped simultaneously. The acceptable values are from two (a single pair swapped) to eight (four pairs swapped) Optimizer swaps.
swap_rate	Sets the maximum number of swaps that Optimizer is allowed to make in a single day. This parameter is only relevant if swap_mode is set to AUTO. Note that reflects the total number of devices to be swapped in a day. A value of 24 would allow 12 pairs of devices to be swapped in a day.

**Example** The following command file sets up Optimizer to analyze data from the previous seven days and to start figuring out swap suggestions after three days of collecting data. It also sets Optimizer to User Approved mode and sets the maximum number of simultaneous swaps to eight (four pairs of hypervolumes):

```
set control_parms start_mode=AUTO,
min_perf_period=72,
workload_period=168
swap_mode=USER_OK,
max_simult_swaps=8,
swap_rate=50;
```

## Setting advanced parameters

Symmetrix Optimizer provides the following advanced parameters that can be set:

```
set advanced_parms [max_rollback = <max_days>,]
```

```
[hot_spot = <TRUE | FALSE>,];
```

Where:

*max\_days* — Specifies the maximum number of days for rollbacks. For example, if 8 were specified, only swaps that took place within the past eight days would be eligible for rollback.

*hot\_spot* — For each time sample, the disk access time is computed from the statistics collected.

## Setting time windows

Solutions Enabler V7.3 introduces the `symtw` command for managing time windows. You can use this command to convert any existing time windows to the new format. The new time window format is for FAST, FAST VP, and Symmetrix Optimizer.

Refer to [Chapter 7, “Managing Time Windows”](#) for information about managing time windows with the `symtw` command.

### IMPORTANT

You can (and should) use the `symtw convert` command to convert any existing time windows (previously created with the `symoptmz` command) to the new format.

For information about managing time windows for arrays running versions of Enginuity earlier than 5874 Q22011, refer to [Appendix Appendix G, “Managing Legacy Time Windows”](#).

## Setting the swap priority for devices

There may be devices that store critical application data for which you want the highest possible performance, or devices that you never want Optimizer to swap.

Use the following command to set the device priority:

```
set swap_priority <NO_SWAP | NORMAL | HIGH> for
dev SymDevStart[:SymDevEnd];
```

Where:

**NO\_SWAP** — sets the device to never be swapped.

**NORMAL** — allows swaps that will improve performance.

**HIGH** — requires the best performance.

### Example

To set the swap priority for device 020 to **HIGH**, commit a command file containing the following:

```
set swap_priority HIGH for dev 020;
```



## Setting manual swap lists

### Enginuity 5874 and higher

To set a manual swap list, use the following form:

```
set dev_swap SymDevName1 with SymDevName2
[, SymDevName3 with SymDevName4, ...]
[begin_at=TimeVal];
```

#### Example

```
set dev_swap 0030 with 0040;
```

**Note:** Solutions Enabler V7.1.1 and higher supports manual swaps between different device protection types.

### Enginuity 5773 and earlier

To set a manual swap list, use the following form:

```
set swap_list {Hyper1} with {Hyper2} [, {Hyper3} with {Hyper4},... ]
[begin_at=TimeVal];
```

Where:

*Hyper<sub>n</sub>* — Is of the form DDD,I,T,HH:

DDD — Is the director identifier

I — Is the director interface

T — Is the target ID

HH — Is the hyper number

*TimeVal* — Is in the form of MMDDYYYY:HHMMSS.

## Reviewing Optimizer plans

This section provides examples of the Optimizer-generated plans for swaps.

### Show parameters

To show Optimizer's planned activities, use the following arguments with the `symoptmz` command:

```
symoptmz show [-v] -swap_list | -activity_list
               [-manual | -generated ]

show -parms [-dp | -vp] [-offline]
show -composite [-dp | -vp]
show [-v] -swap_list | -rollback_list
```

Where:

`-activity_list` — Displays all the swap lists currently known to Optimizer (interchangeable with `-swap_list`).

`-generated` — Displays Optimizer-generated swap plans (and FAST-generated plans).

`-manual` — Displays any user-defined swap plans.

`-parms` — Displays information about the control parameters of Symmetrix Optimizer.

`-dp` — Filters the display to include only standard devices. (Disk group provisioning)

`-vp` — Filters the display to include only thin devices. (Virtual provisioning)

If neither filter is specified, all data movement time windows display.

`-offline` — Displays the control parameters and time windows from the cached SYMAPI database.

`-composite` — Displays composite time windows. Composite time windows are generated by Optimizer by combining all known user-defined time windows.

`-dp` — Filters the display to include only standard devices.

`-vp` — Filters the display to include only thin devices.

If neither filter is specified, all data movement time windows display.

---

**Note:** With Enginuity 5568 and earlier, the `symoptmz show` command is only supported without any options.

---

`-rollback_list` — Displays a list of possible rollback points.

`-swap_list` — Displays all the swap lists and migrations currently known to Optimizer.

## Showing control parameters

*Example* To show the Optimizer parameters set for Symmetrix 234 for standard devices only, enter:

```
symoptmz -sid 234 show -parms -dp
```

Output similar to the following displays:

Optimizer Control Parameters

. . .

Number of Swap Time Windows : 1

Time Window ID	: The Default Time Window
Type	: Swap
Provisioning	: Standard
Flags	: Exclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

Performance Time Windows : 1

Time Window ID	: The Default Time Window
Type	: Performance
Flags	: Inclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

*Example* To show the Optimizer parameters set for Symmetrix 234 for thin devices only, enter:

```
symoptmz -sid 234 show -parms -vp
```

Output similar to the following displays:

Optimizer Control Parameters

. . .

Number of Swap Time Windows : 1

Time Window ID	: The Default Time Window
Type	: Swap
Provisioning	: Thin
Flags	: Exclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

Performance Time Windows : 1

Time Window ID	: The Default Time Window
Type	: Performance
Flags	: Inclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

*Example* To show the Optimizer parameters set for Symmetrix 234 for all devices, enter:

```
symoptmz -sid 234 show -parms
```

Output similar to the following displays:

Optimizer Control Parameters

. . .

Number of Swap Time Windows : 2

Time Window ID	: The Default Time Window
Type	: Swap
Provisioning	: Standard
Flags	: Exclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

Time Window ID	: The Default Time Window
Type	: Swap
Provisioning	: Thin
Flags	: Exclusive
Periodicity	: Once
Start Date	: None
Stop Date	: None

Performance Time Windows : 1

Time Window ID	: The Default Time Window
Type	: Performance
Flags	: Inclusive
Periodicity	: Once
Start Date	: None

Stop Date : None

**Example** To show the Optimizer parameters set for Symmetrix 234 for all devices using the `-offline` option, enter:

```
symoptmz -sid 234 show -parms -offline
```

Output similar to the following displays:

```
symoptmz -sid 234 show -parms -offline
```

Optimizer Control Parameters

. . .

Number of Swap Time Windows : 2

```
Time Window ID      : The Default Time Window
Type                : Swap
Provisioning        : Standard
Flags               : Exclusive
Periodicity         : Once
Start Date          : None
Stop Date           : None
```

```
Time Window ID      : The Default Time Window
Type                : Swap
Provisioning        : Thin
Flags               : Exclusive
Periodicity         : Once
Start Date          : None
Stop Date           : None
```

Performance Time Windows : 1

```
Time Window ID      : The Default Time Window
Type                : Performance
Flags               : Inclusive
Periodicity         : Once
Start Date          : None
Stop Date           : None
```

## Showing composite windows

**Example** To show all the composite windows for Symmetrix 234, enter:

```
symoptmz -sid 234 show -composite
```

Output similar to the following displays:

Number of Composite Time Windows : 3

Start Time	Stop Time	Type	Prov.	Flags
-----	-----	----	-----	-----
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Swap	STD	Exclusive
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Swap	THIN	Exclusive
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Perf	N/A	Inclusive

*Example* To show the composite windows for Symmetrix 234 using the `-dp` option, enter:

```
symoptmz -sid 234 show -composite -dp
```

Output similar to the following displays:

Number of Composite Time Windows : 2

Start Time	Stop Time	Type	Prov.	Flags
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Swap	STD	Exclusive
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Perf	N/A	Inclusive

*Example* To show the composite windows for Symmetrix 234 using the `-vp` option, enter:

```
symoptmz -sid 234 show -composite -vp
```

Output similar to the following displays:

Number of Composite Time Windows : 2

Start Time	Stop Time	Type	Prov.	Flags
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Swap	THIN	Exclusive
Dec 31 19:00:00 1999	Dec 30 19:00:00 2030	Perf	N/A	Inclusive

## Showing the swap list

With Solutions Enabler V7.1.1, the output for the `symoptmz show` command has been updated to include the estimated percent of the plan complete and the estimated time to completion.

*Example* The following command displays the swap list for Symmetrix 234. The new fields in this display are bold in the output and described with the other field descriptions after the example:

```
symoptmz show -swap_list -sid 234
```

This operation may take up to a few minutes. Please be patient...

```
Symmetrix ID           : 000194900234
Number of Plans        : 2

Plan ID                : 12222009:130114
Plan Type              : Manual Defined
Plan State             : ApprovedWithConfigParameters
Start Time             : Tue Dec 22 21:30:23 2009
Percent Complete       : 90%
Estimated time to completion : 00:02:15
Number of Groups       : 1
```

```
Group 1:
{
  Group Attributes      : Optimizer Manual Swap
  Group State          : InProgress
  Time Started         : Tue Dec 22 21:30:23 2009
  Time Completed       : N/A

  Percent Complete     : 90%
  Estimated time to completion : 00:02:15
```

```

Swap Pairs (2)
{
    Source Device                                Target Device
    -----
    Dsk
    Sym  Grp Group Name      Prot      Sym  Grp Group Name      Prot
    ----  -  -
    0020  004 sata_disks      R5 (3+1)  0086  004 sata_disks      R5 (3+1)
    0031  004 sata_disks      R1        0055  004 sata_disks      R1
    }
}

Plan ID                : 12222009:154359
Plan Type              : Auto Generated
Plan state             : InProgress
Start Time             : Tue Dec 22 20:30:23 2009
Percent Complete       : 5%
Estimated time to completion : 04:12:30
Number of Groups       : 3

Group 1:
{
    Group Attributes      : Optimizer Generated
    Group State           : InProgress
    Percent Complete      : 20%
    Estimated time to completion : 00:40:12
}

. . .

```

The terms in the display are explained in [Table 25](#).

**Table 25** New fields in Optimizer display (page 1 of 2)

Field	Description
Plan Type	Defines where the plan originated. Possible values are: <ul style="list-style-type: none"> <li>Manual Defined — User defined swap/rollback plan.</li> <li>Auto Generated — Optimizer/FAST auto-generated plan.</li> </ul>
Plan State	Defines the current state of the plan. The possible values are: <ul style="list-style-type: none"> <li>NotApproved — The plan is not scheduled.</li> <li>ApprovedWithConfigParamaters — The plan is approved and scheduled according to a set time window.</li> <li>ApprovedWithDelay — The plan is approved and scheduled according to a user-defined start time.</li> <li>Validating — The Optimizer server is validating the manual swap/rollback plan.</li> <li>ConfigInProgress — Data movement is in progress for the plan.</li> <li>Aborting — The plan is in the process of aborting.</li> </ul>
Percent Complete	Provides an estimate of how much (percent) of the plan has completed.
Estimated time to completion	Provides an estimated time to the plan's completion based on the invalid track data and timestamps returned.
Delayed Start	Indicates if there is a delay in the plan start time. Possible values are: <ul style="list-style-type: none"> <li>None — No delay.</li> <li>Time — Displays an additional Start_time field.</li> </ul>
Group Attribute	Indicates where the plan originated. Possible values are: <ul style="list-style-type: none"> <li>Optimizer Manual Swap</li> <li>Optimizer Manual Rollback</li> <li>Optimizer Generated</li> <li>Fast Generated</li> </ul>

**Table 25** New fields in Optimizer display (page 2 of 2)

Field	Description
Group State	Displays the state of the group. The possible values are: <ul style="list-style-type: none"> <li>• Done</li> <li>• InProgress</li> <li>• NotStarted</li> <li>• Failed — The execution of the entire group has failed and will be retried.</li> </ul>
Time Started	The time the swap started.
Time Completed	The time the swap completed.

*Example* To show the swap list output for Symmetrix 234 with the verbose (-v) option, enter:

```
symoptmz show -swap_list -sid 234 -v
```

```
Symmetrix ID           : 000194900234
Number of Plans        : 2

Plan ID               : 12222009:130114
Plan Type             : Manual Defined
Plan State            : ApprovedWithConfigParameters
Start Time            : Tue Dec 22 21:30:23 2009
Percent Complete      : 90%
Estimated time to completion : 00:02:15
Number of Groups      : 1
```

```
Group 1:
```

```
{
  Group Attributes      : Optimizer Manual Swap
  Group State           : InProgress
  Time Started          : Tue Dec 22 21:30:23 2009
  Time Completed        : N/A
  Percent Complete      : 90%
  Estimated time to completion : 00:02:15
}
```

```
. . .
```

```
Plan ID               : 12222009:154359
Plan Type             : Auto Generated
Plan State            : CnfigInProgress
Start Time            : Tue Dec 22 20:30:23 2009
Percent Complete      : 5%
Estimated time to completion : 04:12:30
Number of Groups      : 3
```

```
. . .
```

```
Group 3:
```

```
{
  Group Attributes      : FAST Generated(Compliance)
  Group State           : NotStarted
  Time Started          : N/A
  Time Completed        : N/A
  Percent Complete      : 0%
  Estimated time to completion : 04:12:30
}
```

```
Move Devices(6)
```

```
{
  SRC Device(s)         : 0042 0043 0044
  SRC Protection Type    : R6(14+2)
  SRC Storage Group Name : OraSales
  SRC Tier Name          : N/A
}
```

```

SRC Disk Group Number      : 3
SRC Disk Group Name        : fiber_disks
SRC Protection Type        : R5(7+1)
TGT Tier Name              : ArchiveDBTier
TGT Disk Group Number      : 4
TGT Disk Group Name        : sata_disks
TGT Disks                  : [16A, D, 1] [01A, C, E] [15C, C, D] [15D, D, A]
                           [01C, D, 2] [12A, D, D] [15B, C, E] [07A, C, 2]
                           [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]
                           [01F, C, 5] [12H, C, 1] [07H, D, 1] [07F, D, 8]

Percent Complete           : 0%

SRC Device(s)              : 0059 005A 005B
SRC Protection Type        : R5(3+1)
SRC Storage Group Name     : OraSales
SRC Tier Name              : PrimeDBTier
SRC Disk Group Number      : 2
SRC Disk Group Name        : flash_disks2
TGT Protection Type        : R5(3+1)
TGT Tier Name              : WorkDBTier
TGT Disk Group Number      : 3
TGT Disk Group Name        : fiber_disks
TGT Disks                  : [16C, D, 1] [01B, C, E] [15A, C, D] [15C, D, A]
                           [01A, D, 2] [12B, D, D] [15C, C, E] [07B, C, 2]
                           [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]

Percent Complete           : 0%
}
}

```

**Example** To show the manual swap list for Symmetrix 234, enter:

```
symoptmz show -swap_list -manual -sid 234
```

Output similar to the following displays:

```

Symmetrix ID               : 000194900234

Plan ID                    : 12222009:130114
Plan Type                  : Manual Defined
Plan State                 : ApprovedWithConfigParameters
Start Time                 : Tue Dec 22 21:30:23 2009
Percent Complete           : 90%
Estimated time to completion : 00:02:15
Number of Groups           : 1

Group 1:
{
Group Attributes           : Optimizer Manual Swap
Group State                : InProgress
Time Started               : Tue Dec 22 21:30:23 2009
Time Completed             : N/A
Percent Complete           : 90%
Estimated time to completion : 00:02:15

Swap Pairs (2)
{
Source Device              Target Device
-----
Dsk
Sym  Grp Group Name        Prot      Sym  Grp Group Name        Prot
-----
0020  004 sata_disks        R5(3+1)  0086  004 sata_disks        R5(3+1)
0031  004 sata_disks        R1       0055  004 sata_disks        R1
}
}

```



**Example** To display a list of Optimizer-generated plans for Symmetrix 234, enter:

```
symoptmz show -swap_list -sid 234 -generated
```

Output similar to the following displays:

This operation may take up to a few minutes. Please be patient...

```
Symmetrix ID           : 000194900234
```

```
Plan ID                : 12222009:154359
Plan Type              : Auto Generated
Plan State             : InProgress
Start Time             : Tue Dec 22 20:30:23 2009
Percent Complete       : 5%
Estimated time to completion : 04:12:30
Number of Groups       : 3
```

Group 1:

```
{
Group Attributes       : Optimizer Generated
Group State           : InProgress
Time Started          : Tue Dec 22 21:30:23 2009
Time Completed        : N/A
Percent Complete      : 20%
Estimated time to completion : 00:40:12
```

Swap Pairs (2)

Source Device				Target Device			
Dsk Sym	Grp	Group Name	Prot	Dsk Sym	Grp	Group Name	Prot
0023	004	sata_disks	R5(3+1)	0088	004	sata_disks	R5(3+1)
0032	004	sata_disks	R1	0058	004	sata_disks	R1

Group 2:

```
{
Group Attributes       : FAST Generated(Performance)
Group State           : NotStarted
Time Started          : Tue Dec 22 20:30:23 2009
Time Completed        : N/A
Percent Complete      : 0%
Estimated time to completion : 01:35:31
```

Swap Pairs (2)

Source Device				Target Device			
Storage Sym	Tier	Name	Prot	Storage Sym	Tier	Name	Prot
0045	PrimeTier	R1	OraSales	00E0	PrimeDbTier	R5(3+1)	OraSales
0046	PrimeTier	R1	OraSales	00E1	PrimeDbTier	R5(3+1)	OraSales

Group 3:

```
{
Group Attributes       : FAST Generated(Compliance)
Group State           : NotStarted
Time Started          : Tue Dec 22 21:30:23 2009
```

```

Time Completed           : N/A
Percent Complete         : 0%
Estimated time to completion : 04:12:30

```

```
Move Devices (6)
```

```

{
Source Device           Target
-----
Storage
Sym  Group Name        Tier Name        Prot      Grp Group Name
-----
0042 OraSales          ArchiveTier      R5 (7+1)   004 sata_disks
0043 OraSales          ArchiveTier      R5 (7+1)   004 sata_disks
0044 OraSales          ArchiveTier      R5 (7+1)   004 sata_disks
0059 OraSales          WorkDbTier      R5 (3+1)   001 fiber_disk
005A OraSales          WorkDbTier      R5 (3+1)   001 fiber_disk
005B OraSales          WorkDbTier      R5 (3+1)   001 fiber_disk
}
}

```

---

**Note:** FAST-generated plans will also display in the Optimizer output. There are no plans for thin tiers.

---

## Approving a swap list

User approval mode allows you to see which swaps will occur before they take place. To set Optimizer to user approval mode, use the `set control_parms swap_mode=USER_OK;` line in the command file. Once this is set, Optimizer recalculates a swap list approximately once every hour as long as samples have been collected for the specified minimum performance period. At any time, the latest swap list can be retrieved using the `symoptmz show -swap_list` command. The swap list can then be approved or declined using the following syntax in the command file:

```

set swap <APPROVE | DECLINE>
    [begin_at=TimeVal,]
    TIMESTAMP=TimeVal;

```

The time stamp specified must be the time stamp returned by the last `symoptmz show -swap_list` command. If the command to approve or decline the swap list returns an error, the swap list is probably out of date and a new one is available from the service processor. The latest swap list should then be retrieved again with the `symoptmz show -swap_list` command.

[“Reviewing Optimizer plans” on page 297](#) provides examples of output from the `symoptmz show -swap_list` command.

## Performing rollbacks

Rollbacks are used to undo swaps that conflict with your business rules. Rollbacks are performed by first showing the rollback list, as follows:

```
symoptmz show -rollback_list
```

Use the following command file syntax to rollback a swap:

```
set swap <APPROVE | DECLINE>
  [begin_at=TimeVal,]
  TIMESTAMP=TimeVal,
  [, ROLLBACK];
```

The rollback feature is an *all or nothing* feature. Optimizer reverses all swaps by going backward from the present to a selected earlier time and undoing each and every swap. Use the User Approved mode as described in [“Approving a swap list” on page 306](#) to undo specific swaps. Rollback swaps can be scheduled for execution according to normal Optimizer policy.

---

**Note:** When Optimizer rolls back a swap, if FAST swaps are in the swap list, they will be affected. For more information about FAST, see [Chapter 4, “Fully Automated Storage Tiering.”](#)

---

## Reading Optimizer logs

To retrieve the Optimizer activity log, including swaps that have been completed, use the following command:

```
symoptmz read -log_type RUNTIME [-start DateTime] [-stop DateTime]
```

To retrieve the Optimizer error log, use the command:

```
symoptmz read -log_type ERROR [-start DateTime] [-stop DateTime]
```

The log that is returned includes DOS line-feed characters that can be removed using `sed` or another text editor:

```
cat opt_log.txt | sed s/^M$// > cleaned_opt_log.txt
```

In a UNIX shell, the ^M should be typed using **CTRL-V CTRL-M**.

In addition, the `symoptmz read` command can be used to determine why a swap was not performed.

## Migrating devices

---

**Note:** The `symoptmz migrate` command is not available on Symmetrix VMAX Family arrays running Engenuity 5874 and higher. For those arrays, use the `symmigrate` command, as explained in [Chapter 6, “Enhanced Virtual LUN Technology.”](#)

---

When devices are moved to a new set of disks, this is called a *migration*. A migration can be used to populate newly added disk drives, or to move devices between high performance and high capacity disks. A migration can only be initiated by a user. When devices are moved to a new location, this is called a *relocation*, and it may be used to balance disk usage or to fill in holes left by deleting devices. Relocations are performed by the Optimizer automatically, as needed. A relocation cannot be initiated by a user.

Symmetrix Optimizer `symoptmz` command has been enhanced with the following features for data migrations:

- ◆ The source devices to be migrated can be specified as a device group.
- ◆ The target disks can be a disk group number.
- ◆ You can optionally specify that unmapped and/or unmasked devices can be used as target devices.

When devices or device group devices are migrated to target disks or disk groups, Optimizer looks for free space on the target disks for the migration. If the target disks do not have any free space, you can specify to use unmapped and/or unmasked devices. These options indicate that source devices can be migrated to space occupied by unmapped and/or unmasked devices on the target disks. When this option is specified, Optimizer will NOT look for free space on the target disks but will only look for matching unmapped and/or unmasked devices.

---

**Note:** The unmapped and unmasked options are not recommended if there is free space on the target disks.

---

As with all device migrations, devices on the target disk list must match the size, emulation, and protection type of the source devices for the migration to succeed. To initiate a migration, use the `symoptmz` command with the following syntax in the command file:

```
migrate
  dev[s] SymDevStart1[:SymDevEnd1]
  [, SymDevStart2[:SymDevEnd2], ...]
  TO disk[s] {diskn} [, {diskn}, ...]
  [unmapped=TRUE] [unmasked=TRUE]
  [begin_at=TimeVal];

migrate
  device_group DgName
  TO disk_group_num #
  [unmapped=TRUE] [unmasked=TRUE]
  [begin_at=TimeVal];

migrate
  device_group DgName
  TO disk[s] {diskn} [, {diskn}, ...]
  [unmapped=TRUE] [unmasked=TRUE]
  [begin_at=TimeVal];
```

```

migrate
  dev[s] SymDevStart1[:SymDevEnd1]
  [,SymDevStart2[:SymDevEnd2],...]
  TO disk_group_num #
  [unmapped=TRUE] [unmasked=TRUE]
  [begin_at=TimeVal];

```

Where:

The `diskn` is of the form `DDD,I,T`:

`DDD` is the director Identifier,

`I` is the Director Interface, and

`T` is the Target ID

The `TimeVal` is in the form of `MMDDYYYY:HHMMSS`.

*Example* An example of a device migration command file follows:

```

migrate devices 240, 245:247 to disks {16B, C, 0}, {4D, C, 0};
migrate device_group Mydgname to disk_group_num 1;
migrate devices 240, 245:247 to disk_group_num 2;
migrate device_group Mydgname to disks {16B, C, 0}, {4D, C, 0};
migrate devices 240, 245:247 to disks {16B, C, 0}, {4D, C, 0} unmapped=TRUE;
migrate device_group Mydgname to disk_group_num 0 unmasked=TRUE;
migrate devices 240, 245:247 to disk_group_num 0 unmapped=TRUE unmasked=TRUE;

```



# CHAPTER 10

## Performing Double Checksum Operations

This chapter describes the Double Checksum concepts and how to minimize the impact of I/O errors on database consistency during I/O transfers between hosts and Symmetrix storage devices using the Double Checksum commands of the SYMCLI. This chapter covers the following topics:

◆ Overview.....	312
◆ Implementing EMC Double Checksum for Oracle .....	313
◆ Implementing Generic SafeWrite for generic applications .....	317
◆ Syntax and examples .....	322

## Overview

The EMC Double Checksum feature provides a method to help minimize the impact of I/O errors on database consistency during I/O transfers between hosts and Symmetrix storage devices.

For Oracle, **EMC Double Checksum for Oracle** contains a rich set of checks that can be natively performed by the Symmetrix. For each Relational Database Management System (RDBMS) write in the Symmetrix system, checksum values are computed and compared to test the data for any corruption picked up along the way from the host. Although errors of this kind are infrequent, they can have a considerable effect on data availability and recovery. Refer to [“Implementing EMC Double Checksum for Oracle” on page 313](#) for details on this feature.

For generic RDBMS applications, **EMC Double Checksum for Generic Applications** provides the Generic SafeWrite<sup>1</sup> feature to help protect critical applications from incurring an incomplete write, and subsequent torn page, due to a failure with a component connected to the Symmetrix Front End Channel Adapter. Generic SafeWrite is most often used to protect against corruption are from HBA and link failures including server crashes, where essentially, it will help protect against fractured writes that can occur before the data reaches the Symmetrix array. Refer to [“Implementing Generic SafeWrite for generic applications” on page 317](#) for details on this feature.

This chapter contains overview and concept information; see the `symchksum` command in the *EMC Solutions Enabler Symmetrix CLI Command Reference* for the complete list of syntax and options.

### Traditional methods of preventing data corruption

Data corruption checking is an integral part of most RDBMS products. For instance, Oracle computes a checksum that verifies the data within each page. If corruption occurs, the checksum will be incorrect when the data is read. However, this checking only takes place within the host system—not the storage array.

As a result, if there is corruption after the data has left the host system, it will not be detected until that data is read back into the system, which can be some time—maybe months—later. The RDBMS will issue an alert, and then the data must be rebuilt from backups and database logs. While a corruption remains undetected, the number of database logs required for recovery increases. This causes the data recovery process to be more complex and time-consuming.

### Data corruption between host and conventional storage

Although data appears to the host to travel directly to the Symmetrix array, it passes through multiple hardware and software layers. These can lead to problems such as corruption introduced by errors in the operating system or the I/O driver. Hardware can also introduce corruption, such as errors in the host adapter, cable and connector problems, static electricity, and RF noise and interference.

---

1. Generic SafeWrite has been created with RDBMS in mind. Applications intended for use with Generic SafeWrite include, but are not limited to Microsoft Exchange, Microsoft SQL Server, DB2/UDB and Oracle. Generic SafeWrite is not intended for use with applications where torn pages are not a concern, such as fileshares or FTP servers.



This means that valid data within the RDBMS might arrive corrupted at the storage device. The storage device writes the data as is, because it has no way of validating the data.

## Benefits of checking within the Symmetrix array

With this feature, the Symmetrix array can perform error checks on data pages handled within a checksum extent *as they are written to the disk*. The check takes place before the write command is acknowledged. If an error is detected within the blocks of the extent, the I/O can be rejected or reported in a phone home connection or logged in the Symmetrix error log facilities. The error action can be specified by the administrator.

This checking feature minimizes the possibility of data corruption occurring between the host and the Symmetrix array. It improves the recovery time by flagging the error at the time of the "write." When this error condition is raised, and reject I/O is selected, Oracle takes an action, such as taking the tablespace offline.

## Implementing EMC Double Checksum for Oracle

The `symchksum` command allows you to perform control operations that manage the checksum I/O extents.

For example, to enable Oracle checksum checking on the extents of all the devices that define the current database instance and then to phone home on error, enter:

```
symchksum enable -type Oracle -phone_home
```

This command requires an Oracle PAK license.

The following are current restrictions for this feature:

- ◆ Oracle databases version 8.0 and above are supported.
- ◆ Data block size is limited to 32 KB.
- ◆ Checksum data objects can only be Oracle data files, control files, and redo logs.
- ◆ Devices cannot be diskless devices (DLDEVs).

The Oracle instance being checked must be started with the following `init.ora` configuration parameter set to true: `db_block_checksum=true`.

---

**Note:** Manipulating remote devices over an SRDF link is not supported. The `symchksum` command can only manipulate devices in locally attached Symmetrix arrays.

---

## Other checksum operations

The following additional checks are available:

- ◆ **MagicNumber** — Verifies the 3-bit magic number that appears in Oracle data blocks. (Enabled by default.)
- ◆ **NonzeroDba** — Checks for non-zero data block address. The dba stored in a data block is never zero. (Enabled by default.)
- ◆ **Check\_All\_Blocks** — Applies checks to each block in a write. Otherwise, only the first block is checked.
- ◆ **Straddle** — Checks that the write does not straddle known Oracle areas.

- ◆ **Check\_dba** — Checks that the logical address embedded by Oracle is compatible with the storage address of the writes.

**Note:** If `-check_dba` is selected, extents from separate datafiles cannot be collapsed. The `-check_dba` option requires data specific to each datafile, which would be lost in an extent collapse.

As a result, each datafile will occupy at least one Symmetrix device extent. When using a logical volume manager and `-check_dba`, this creates a limitation of 200 datafiles per volume group or disk group.

To enable Oracle checksum and add the optional operations of check DBA and check all blocks, enter:

```
symchksum -type Oracle enable -check_dba -check_all_blocks
```

With the addition of these new tests, the output when you list the extents will look similar to the following:

Symmetrix ID: 000187900671

```

D E V I C E S      W I T H      C H E C K S U M      E X T E N T S

                                     Action  Checks
                                     R P C  C  A N  D
                                     e h  h M h S l Z F i
                                     j o  k a k t l   r s
                                     L e n  s g D r B D a c
                                     o c e  u i B a l B c r
Device Name                Num Blk    g t H  m c A d k A t d
                        Dev Exts Siz Type
-----
/dev/sdj                  0048     1 16b Oracle  X X X  X X . . . X X X
/dev/sdz                  0058     1 16b Oracle  X X X  X X . . . X X X

```

2 Devices with Checksum extents were found.

You can use this output to determine which features are enabled on the devices with checksum extents.

## Suppressing default features

By default, the `symchksum` command attempts to select the following tests:

- Block checksum value (Checksum)
- Blocksize verification (BlkSize)
- 3-bit magic number (MagicNumber) in Oracle data blocks (not available for redo logs)
- Checks for non-zero data block addresses (NonZeroDba)

To turn off any of the automatic checksum features, use the `-suppress_feature` option and supply the name of the feature, for example:

```
symchksum -type Oracle enable -suppress_feature MagicNumber
```

The `-suppress_feature` option is only for the operations that run by default. If you want to turn off an option that was manually enabled, such as `-phone_home`, you must disable the checksum operation and begin again with a new `symchksum enable` command.

## Enabling checksum options

When using the `symchksum enable` command, you can decide if you would like to reject the I/O, or have the Symmetrix array phone home when a checksum error is detected.

If an I/O is not a multiple of the object block size, the user can choose to reject the I/O. This is called a fractured I/O, and is selected with the `-fractured_reject_io` option. When using this option, the `-reject_io` option must also be used.

When extents are enabled with the `-discard` option, EMC Double Checksum writes blocks to disk until a failed block is detected. The `-discard` option divides a large I/O into smaller units of 32 KB each. When a checksum failure is detected, all blocks in that unit and subsequent units are discarded. When using this option, the `-reject_io` option must also be used.

The `symchksum enable` command understands the Oracle database structure. The feature can be enabled for tablespaces, control files, redo logs, or the entire database.

---

**Note:** For Oracle 9i and above, if the block size for a tablespace is altered, then the user must disable and then re-enable the extents of the tablespace to ensure that the block size of the enabled extents match the block size of the tablespace.

---



---

**Note:** When FF or power down occurs, extents are lost. You must run the `symchksum enable` command again.

---

## Verifying checksum is enabled

The `symchksum` command also allows you to verify that the data file's extents are currently enabled for checksum checking. This provides an easy way to determine if the specified tablespace or instance is fully protected by the Symmetrix checksum feature. The `verify` action will report if all, some, or none of the Oracle data file's extents are enabled for checksum checking. This is useful in environments where the database configuration changes frequently. An example of this is if a new data file was added but checksum was not enabled for the new file.

The `symchksum verify` command understands the Oracle database structure. The feature can be enabled for tablespaces, control files, redo logs, or the entire database.

## Validating for checksum operations

The `symchksum` command also allows you to validate your Oracle tablespace or instance for checksum operations without performing any active actions. This is helpful when you want to know if your database environment is configured to support Symmetrix checksum functionality without actually making any changes.

If the validate is successful, you can enable EMC Double Checksum on the specified Oracle database or tablespace. The following items are validated:

- ◆ Oracle version 8 or higher is installed
- ◆ Oracle's checksum initialization parameter is set (`db_block_checksum`)

- ◆ If the Oracle data file is created on a striped LVM, that the LVM stripe width is a multiple of the Oracle block size
- ◆ Oracle data file's block size is less than or equal to 32 KB
- ◆ The Symmetrix Engenuity version supports the checksum functionality
- ◆ Each Symmetrix device (prior to Engenuity V5773) has the Checksum flag set
- ◆ Each Symmetrix device has a supportable number of extents defined

The `symchksum validate` command understands the Oracle database structure. The feature can be enabled for tablespaces, control files, redo logs, or the entire database.

## Disabling checksum

The `symchksum disable` command understands the Oracle database structure. The feature can be enabled for tablespaces, control files, redo logs, or the entire database.

The `symchksum disable` command can also be used on a device basis. This capability is not normally used, but is provided in the event the tablespace was dropped before EMC Double Checksum was disabled for that object.

When the `disable` action is specified for a Symmetrix device, the `-force` flag is required. Disabling extents in this way can cause a mapped tablespace or database to be only partially protected, therefore, use this option with caution. All the extents being monitored for checksum errors on the specified Symmetrix device will be disabled.

## Functionality by Engenuity level

In support of the Oracle and EMC Hardware Assisted Resilient Data (HARD) initiative, EMC Double Checksum provides end-to-end database integrity by employing sophisticated error detection technology to detect and prevent corruption of data blocks written to Symmetrix arrays. [Table 26](#) shows the **EMC Double Checksum for Oracle** functionality provided in each Engenuity version

**Table 26 EMC Double Checksum for Oracle functionality by Engenuity level**

Engenuity Level	Block checksum verification	Dial home alert	Discard	Max extents <sup>a</sup> / Symmetrix volume	Magic number detection	Zero dba detection	All blocks option <sup>b</sup>	Straddle <sup>c</sup>
5670 rev 49-	Yes	Yes	Forced On	200	No	No	No	No
5670 rev 50+	Yes	Yes	Yes	200	Yes	Yes	No	No
5670 rev 75.71+	Yes	Yes	Yes	200	Yes	Yes	Yes	Yes

a. Symmetrix extents are internal to the Symmetrix array and are used to map Oracle tablespace locations; they are not the same as Oracle extents

b. For Magic Number Detection and Zero dba Detection, the default mode is to check only the first block in an I/O. If the All Blocks Option is selected in the SYMCLI, then Magic Number Detection and Zero dba Detection checks are performed for each Oracle block. For the checksum check, the only mode is to sum the I/O as a whole; it is not affected by the All Blocks Option.

c. Sd an extension to the Oracle and EMC HARD initiative, EMC Double Checksum functionality checks for I/O that overlaps defined database areas. For example, consider a database that has data from blocks 16 to 105. If an I/O is received for blocks 100 to 200, this I/O is said to straddle the database's data. This is unexpected and likely to be a potentially corrupting I/O. If the Straddle option is selected in the SYMCLI, then EMC Double Checksum will consider straddling I/O to be an error.

# Implementing Generic SafeWrite for generic applications

Generic SafeWrite (GSW) is used to help protect critical applications from incurring an incomplete write, and subsequent torn page, due to a failure with a component connected to the Symmetrix Front End Channel Adapter.

## Torn pages: Using generic safewrite to protect applications

A Relational Database Management System (RDBMS), such as Oracle and Microsoft SQL Server, structure data within database files using pages (also referred to as blocks). Pages within a database are the smallest allocation unit size possible for a database object (such as a table or a row).

For example, the page size for Microsoft SQL Server, and for Oracle, is 8 KB. Although this size is configurable, it is usually set to 8 KB. If an incomplete page is written to a database file, a corruption to the database will occur. The resulting corruption is commonly referred to as a *torn page*.

Torn pages are only detected by most RDBMSs after the corruption has been written, when that area of the database is read, which could be long after when the corruption was introduced. In general, the only recovery from a torn page is to perform a restore from a backup (some RDBMSs allow page-level restores, while others require a complete database restore). Torn pages can occur due to failures in various components that lie between the RDBMS and the storage array. Some of these components include the operating system, file system, logical volume manager, I/O driver, host bus adapter, Fibre or SCSI link and storage adapter.

The EMC Double Checksum Generic SafeWrite feature can help protect critical applications from incurring incomplete writes, and subsequent torn pages, due to a failure with a component connected to the Symmetrix Front End Channel Adapter.

Most often, Generic SafeWrite will be used to protect against corruption that occurs when the HBA or link fails (including server crashes). In this scenario, Generic SafeWrite will protect against fractured writes that occur before the data reaches the Symmetrix array.

## Why generic?

Generic SafeWrite is deemed *generic* because the checks performed to ensure complete data are application independent. For instance, Generic SafeWrite will not perform any Oracle- or SQL Server-specific checksums to verify data integrity. It is important to note that for Oracle, EMC Double Checksum for Oracle provides a rich set of checks which can be natively performed by the Symmetrix array. For more information on EMC Double Checksum for Oracle, refer to [“Implementing EMC Double Checksum for Oracle” on page 313](#).

## Where to enable Generic SafeWrite

Generic SafeWrite only needs to be enabled for specific devices on the Symmetrix array. For a RDBMS, Generic SafeWrite only needs to be enabled for devices that support data files. The list below gives an example of database files where the supporting devices for these files should have Generic SafeWrite enabled:

Microsoft Exchange:

- .edb files

- .stm files

Microsoft SQL Server:

- .mdf files
- .ndf files

Oracle:

- Data files
- Control files

---

**Note:** Manipulating remote devices over an SRDF link is not supported. The `symchksum` command can only manipulate devices in locally attached Symmetrix arrays.

---

It is recommended to enable Generic SafeWrite for database file devices, though it is not necessary to enable it for database log devices. In general, a RDBMS will write to its respective log file with a 512 byte sector alignment. The RDBMS can therefore determine the last sector that was correctly written and subsequently discard or rollback any transactions that are incomplete.

---

**Note:** It is always a best practice to separate the location of database files and log files for a given database onto unique devices. There are cases, however, where the data file and log file may share the same device. In this case, it is still possible to have GSW enabled, however, there will be a performance impact to the log writes which may impact application performance.

---

There are no restrictions regarding the size of a device or the number of devices where GSW can be enabled. However, for both Generic Safewrite and Oracle Safewrite only SRDF/S and TimeFinder/Mirror are supported. It is also supported to enable GSW on filesystems across all logical volume managers as well as on raw devices, given the OS platforms are supported by the Solutions Enabler Storage Resource Management (SRM) component. When using filesystems on Windows and Linux hosts, for performance reasons, it is strongly recommended to ensure the filesystems are properly aligned with the storage. For more information regarding filesystem alignment, refer to *Using diskpart and diskpart to Align Partitions on Windows Basic and Dynamic Disks* available on EMC online support.

## Configuring Generic SafeWrite

To use Generic SafeWrite, you must:

- ◆ For Engenuity versions prior to 5773, enable the `RDB_cksum` device flag on all devices targeted for Generic SafeWrite
- ◆ Run a discover operation to update the physical device information in the SYMAPI database
- ◆ Enable Generic SafeWrite on all devices targeted for Generic SafeWrite use with the `symchksum` command

## Setting the RDB\_cksum Symmetrix device flag

---

**Note:** Symmetrix arrays running Solutions Enabler V7.0 and higher do not require the RDB\_cksum device flag enabled for Double Checksum and Generic SafeWrite use.

---

For Engenuity versions prior to 5773 only:

Before you can use Generic SafeWrite, the RDB\_cksum Symmetrix device flag must be enabled on all devices targeted for Generic SafeWrite use. This change does not turn Generic SafeWrite on, it only allows it to be enabled on the specified devices.

The RDB\_cksum device flag can be set by using the SYMCLI `symconfigure` command, which will perform a Symmetrix configuration change. For more information on using `symconfigure`, refer to [Chapter 1, “Managing Configuration Changes.”](#)

---

**Note:** If `symconfigure` cannot be used, the appropriate device flag can also be set on the array by a EMC Customer Support Engineer.

---

The following is an example command:

```
symconfigure -sid 54 -f c:\enable_cksum.txt commit
```

Where the `c:\enable_cksum.txt` file contains the following command:

```
set device 0015:0019 attribute=RDB_Cksum;
```

---

**Note:** When using `symconfigure`, if metavolumes are used, this flag only the metahead needs to be specified.

---

## Enabling Generic SafeWrite

Once the device flags are set on the Symmetrix array, it is then possible to use the `symchksum` command to enable Generic SafeWrite. Before running the `symchksum` command, confirm the following:

- ◆ The devices that you enabled for Generic SafeWrite are visible to the host from where the `symchksum` command will be run.
- ◆ Run a `symcfg discover` command after presenting devices to a host, in order to update the SYMAPI configuration database with the correct physical drive information.

Using the `symchksum` command, Generic SafeWrite can be enabled by specifying a specific device, a range of devices, or a device group.

### Enabling for a device

To enable Generic SafeWrite for a device, use the command syntax shown in the example below:

```
symchksum enable -type generic dev 005 -sid 54
```

---

**Note:** If this is a metadvice, only the metahead needs to be specified.

---

### Enabling for a range of devices

To enable Generic SafeWrite for a contiguous range of devices:

```
symchksum enable -type generic -devs 005:025 -sid 54
```

## Enabling for a device group

To enable Generic SafeWrite for a device group:

```
symchksum enable -type generic -g sql_data -sid 54
```

---

**Note:** Enabling Generic SafeWrite on a Composite Group (CG) is currently not supported.

---

The `symchksum enable -type generic` command automatically sets the Log, Phone Home, and Generic Double Checksum options as described below:

- ◆ Log — Indicates that errors will be sent to the Symmetrix error log. These events should be visible with the `symevent` command.
- ◆ Phone Home — Indicates that an error will initiate a call by the Symmetrix to EMC Customer Service.
- ◆ Generic — Allows two functions to be performed by the Symmetrix array. First, when an incomplete write is detected, it will be rejected and the Symmetrix will force the I/O to be retried from the host. Then, if the host is not available to retry the I/O, the write will be discarded, preventing it from being written to disk.

## How to disable Generic SafeWrite

Generic SafeWrite can be disabled using the `symchksum disable -type generic` command as shown in the same examples below.

### Disabling for a device

To disable Generic SafeWrite for a device, use the command syntax shown in the example below:

```
symchksum disable -type generic dev 005 -sid 54
```

---

**Note:** If this is a metadvice, only the metahead needs to be specified.

---

### Disabling for a range of devices

To disable Generic SafeWrite for a contiguous range of devices:

```
symchksum disable -type generic -devs 005:025 -sid 54
```

### Disabling for a device group

To disable Generic SafeWrite for a device group:

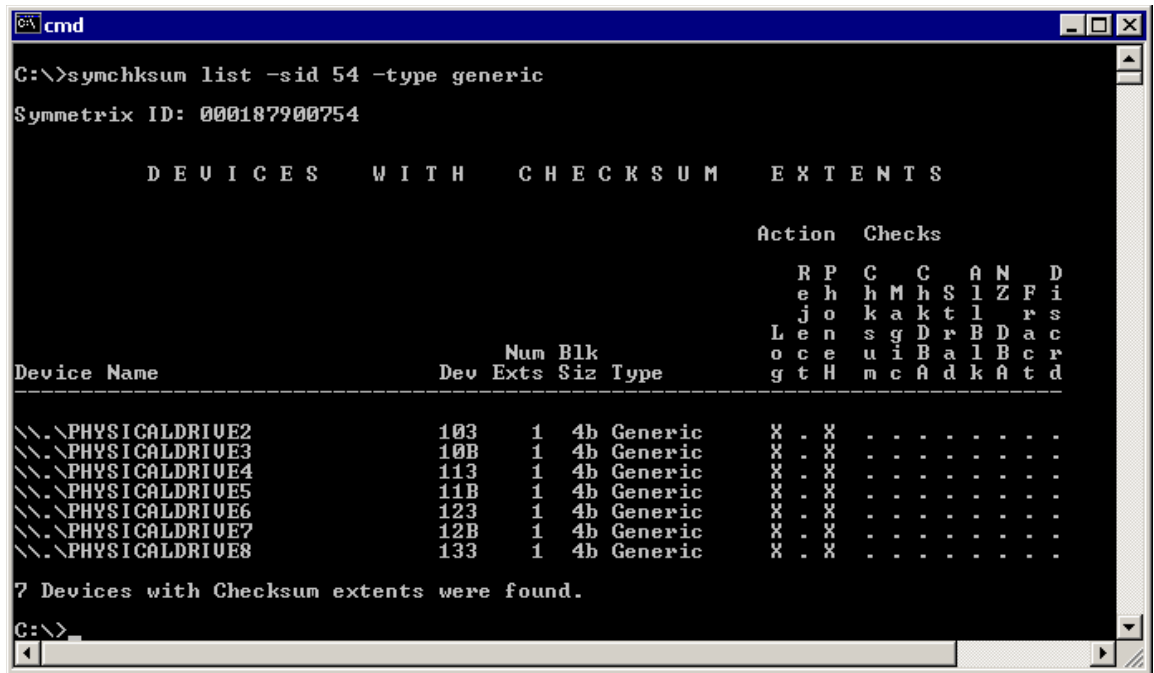
```
symchksum disable -type generic -g sql_data -sid 54
```

## Listing Generic SafeWrite devices

To list which devices are Generic SafeWrite enabled, use the `symchksum list` command with the `-type generic` option. Only Generic SafeWrite-enabled devices that are visible to the host running the `symchksum list` command are returned.



The following example shows the expected output from the list command, with Generic for the type and the Log and PhoneH (short for Phone Home) options set.



```

C:\>symchksum list -sid 54 -type generic
Symmetrix ID: 000187900754

      D E V I C E S      W I T H      C H E C K S U M      E X T E N T S

                                     Action  Checks
                                     R P C C A N D
                                     e h h M h S l Z F i
                                     j o k a k t l r s
                                     L e n s g D r B D a c r
                                     o c e u i B a l B c r
                                     g t H m c A d k A t d

Device Name                               Dev  Exts  Siz  Type
-----
\\.\PHYSICALDRIVE2                        103    1   4b  Generic
\\.\PHYSICALDRIVE3                        10B    1   4b  Generic
\\.\PHYSICALDRIVE4                        113    1   4b  Generic
\\.\PHYSICALDRIVE5                        11B    1   4b  Generic
\\.\PHYSICALDRIVE6                        123    1   4b  Generic
\\.\PHYSICALDRIVE7                        12B    1   4b  Generic
\\.\PHYSICALDRIVE8                        133    1   4b  Generic

7 Devices with Checksum extents were found.
C:\>

```

Figure 30 Generic SafeWrite list output

The `symchksum show` command can also be used to look at a specific device. For example:

```
symchksum show dev 103 -type generic -mb -sid 54
```

## Performance considerations

Application performance should remain unaffected because database log writes, as well as database reads, will be performed normally. However, outside of application performance, there may be a slight increase in the write response time to the database file devices depending on application profile and usage. In general, this response time increase should not impact application performance. Writes to a database file are done asynchronously, therefore write response times to this file are less of a concern than to the log device.

## Syntax and examples

This section contains the `symchksum` argument descriptions and several examples of using the SYMCLI `symchksum` command. See the *EMC Solutions Enabler Symmetrix CLI Command Reference* for the complete list of syntax and options.

**Table 27** `symchksum` syntax and descriptions

Command	Argument	Description
<code>symchksum</code>	<code>list</code>	Lists all devices that currently have checksum checking enabled. The list can be filtered by using the <code>-type</code> option (to show Oracle or generic).
	<code>show</code>	Displays the extents of a specified device that are having checksum checking performed.
	<code>enable</code>	Enables checksum checking on the extents of the specified devices.
	<code>disable</code>	Disables checksum checking on the extents of the specified devices.
	<code>validate</code>	Validates that a specified database or tablespace is able to have checksum checking enabled.
	<code>verify</code>	Verifies whether the specified database or tablespace has checksum checking enabled on all their devices.

To list the devices on Symmetrix array 3890 that have extents being checked for checksum errors, enter:

```
symchksum list -sid 3890
```

To show all the extents of Symmetrix device 0A1 on Symmetrix array 3890 that are being checked for checksum errors, enter:

```
symchksum show dev 0A1 -sid 3890
```

To enable Checksum on the extents of all the devices that define the current database instance and then to phone home on error, enter:

```
symchksum enable -type Oracle -phone_home
```

To enable Checksum on the extents of all the devices that define the tablespace and then to log on error, enter:

```
symchksum enable -type Oracle -tbs SYSTEM
```

To verify that Oracle tablespace USER01 has Checksum enabled on all the devices that have defined it, enter:

```
symchksum verify -type Oracle -tbs USER01
```

To disable Checksum on the current database instance, enter:

```
symchksum disable -type Oracle
```

---

**Note:** Disable by device should only be used under special circumstances. For example, this option can be used to remove extents if a database or a tablespace has been dropped without first doing a normal disable. In this case, disable by device can be used to remove the extents.

---

# CHAPTER 11

## Device Masking

This chapter describes the device masking concepts and how to confine host access to Symmetrix devices using the device masking commands of the SYMCLI. The chapter covers the following topics:

◆ Device masking overview .....	324
◆ Supported topologies .....	327
◆ Using the device masking database .....	329
◆ Configuring device masking .....	329
◆ Discovering host HBAs .....	332
◆ Adding masked devices .....	334
◆ Removing masked devices .....	336
◆ Maintaining the device masking database.....	338
◆ Viewing the VCMDB.....	340
◆ Managing HBA initiators.....	346
◆ Managing the Fibre Channel-to-host interface .....	348

## Device masking overview

This chapter provides device masking information for Symmetrix arrays running Enginuity 5773 and earlier.

---

**Note:** Administrators managing Symmetrix VMAX Family with Enginuity 5874 should not use the `symmask` and `symmaskdb` commands described in this chapter. To mask devices on Symmetrix VMAX Family arrays, refer to [Chapter 3, “Masking Devices with Auto-provisioning Groups.”](#)

---

SYMCLI device masking provides the ability to assign and mask access privileges of host bus adapters (HBAs) to Symmetrix directors and devices by associating one or more devices with an HBA-to-FA connection that you define in the Symmetrix-based device masking database, known as the VCMDB. The VCMDB maintains all access records for an array and the Symmetrix array monitors host access to resolve any conflicts that might arise from multiple hosts having visibility to the same devices.

The device masking commands can also be used to configure heterogeneous hosts with shared access to the same FA port, which is useful in an environment with different host types. However, you can also use Fibre Channel ID lockdown security to protect an HBA from predatory WWN spoofing.

---

**Note:** For a detailed introduction to Solutions Enabler, SYMCLI, and the Symmetrix array, refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

---

### How Symmetrix device masking works

If you define masked channels that allow multiple hosts to connect to a single Symmetrix director, potential conflicts may arise as multiple hosts have access to discover and use the same set of devices. Device Masking allows you to control host access to a set of devices by maintaining a set of entries in the VCMDB on the array that defines the relationship between masked connections and devices.

Each entry includes a host's HBA identity (the HBA port WWN), its associated FA port, and a range of devices mapped to the FA port that should be visible only to the corresponding HBA. Once you make this VCMDB entry and activate the configuration, the Symmetrix makes visible to a host those devices that the VCMDB indicates are available to that host's initiator WWN through that FA port.

The VCMDB on each Symmetrix array specifies the devices that a particular host can access through a specific director. Each director can control access to as many as 256 unique WWNs or 512 iSCSIs (beginning with Enginuity version 5771). As many as 128 fiber director ports, and 64 multi-protocol (iSCSI) ports (depending on the Symmetrix model) can be configured within the device masking VCMDB.

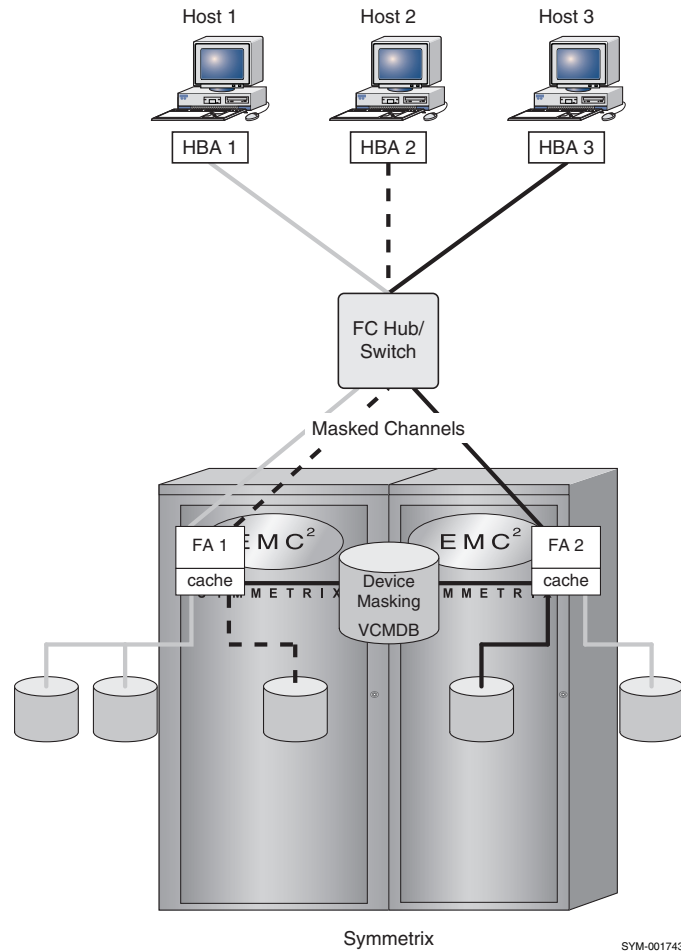
You can initialize, back up, and restore this database. In addition, you can list, add, and remove database entries, clear the database, and manage WWN and iSCSI names.

All Fibre Channel ports connected to the host must be VCM-enabled. Refer to [“Setting port characteristics” on page 103](#).

## Host access examples

When a host attempts to access a Symmetrix storage device, as shown in [Figure 31](#), the host HBA initiator name (supplied when the host logs in to the fabric or arbitrated loop) is passed to the Symmetrix director port.

The Symmetrix records the connection, stores the initiator name in a *login history table* in its memory, and thereafter grants access to only the devices that are available to that initiator through that director port (as specified in the device masking VCMDB). However, if CHAP authentication is enabled in a native iSCSI topology, the VCMDB will first check the credential and secret before granting access.



**Figure 31** Device masking solution with WWN initiators

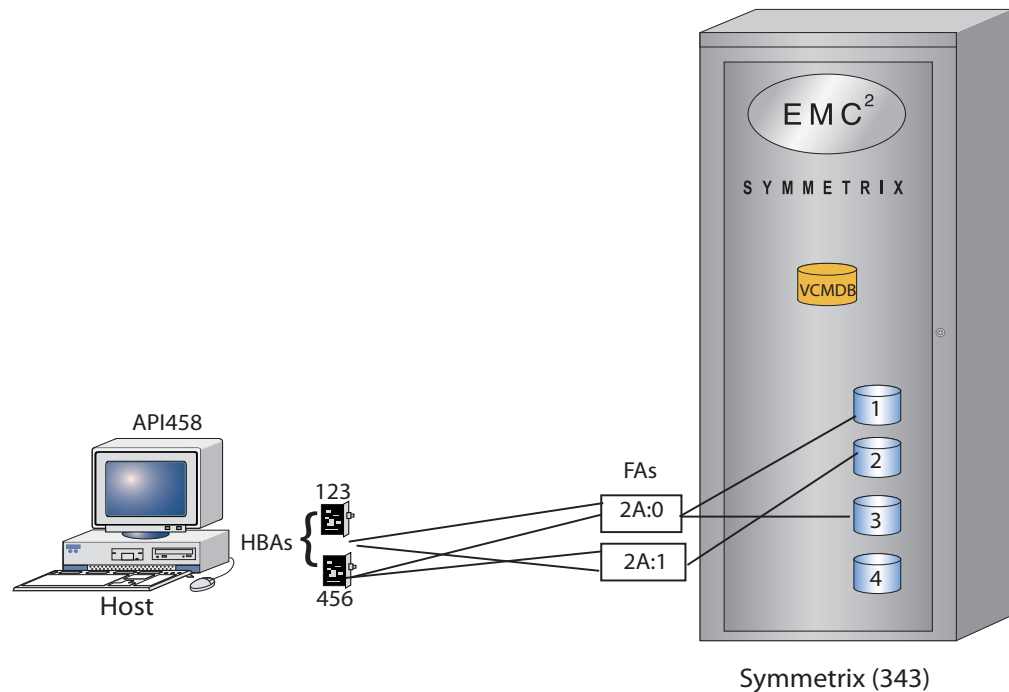
The host HBA port then sends I/O requests directed at particular Symmetrix devices to the director port. Each request includes the identity of the requesting HBA (from which its WWN or iSCSI can be determined) and the identity of the requested device, with its director and logical unit number (LUN).

The software that runs on a Symmetrix system processes each I/O request to verify that the HBA is allowed to access that device. Any request for a device that an HBA does not have access to, returns an error to the host.

In [Figure 31](#), device masking grants Host 1 access to two of the three devices available through FA1, and grants Host 2 access to a third device. Similarly, Host 3 is granted access to only one of the two devices available through FA2, reserving the second device for use as a spare.

Solutions Enabler 7.0 provides the `-host` option to allow the host alias name to be used instead of the initiator types (wwn, alias, or iscsi). In addition, the `ALL` parameter has been added to the `-port` and `-director` options.

These additions can reduce the number of commands required to mask devices, as shown in the following example.



For the above setup there are four records in the VCMDB:

```
WWN 123 on Director port 2A:0 with alias api458/123 and devices 1 and 3
WWN 123 on Director port 2A:1 with alias api458/123 and device 2
WWN 456 on Director port 2A:0 with alias api458/456 and devices 1 and 3
WWN 456 on Director port 2A:1 with alias api458/456 and device 2
```

To mask device 4 with one command:

```
symmask -sid 343 -host api458 -dir ALL -p ALL add dev 4
```

## Managing access for unsupported host platforms

Device masking can also manage access for host platforms that are not supported by this release. If a host can log on to the Symmetrix arrays using a Fibre Channel interface, its access can be controlled.

However, device masking cannot automatically discover the WWNs of host HBAs on unsupported platforms. Instead, you must manually set up a record in the database for these hosts.

## Device masking controls overview

SYMCLI commands `symmask` and `symmaskdb` specifically support the device masking control and monitor operations. With these commands, you can define and query the Symmetrix devices that each host's HBA ports are permitted to access.

The `symmask discover` command can be run on both the control station and the managed hosts. The `symmask discover` action locates paths to the device masking database (VCMDB) and assigns alias names (AWWN/AISCSI) to the HBAs residing on the host on which the command is run if they are NULL. You can optionally use the `rename` action to generate aliases to be assigned.

For detailed man page descriptions of these commands and their options, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

## Database device locking

During the execution of the `symmask` or `symmaskdb` commands, the SYMCLI sets a Symmetrix External Lock (SEL) on the Symmetrix where the device masking database (VCMDB) resides. This lock ensures that only one host can make changes to the database at any one point in time.

If during the processing of a `symmask` or `symmaskdb` command, the host fails, or a Ctrl/C is performed in the middle of the command, the lock might not release and could lock out further needed changes or control actions. If a device masking command is interrupted and the lock is not released, future invocations of a device masking command will display the following error message:

```
The operation failed because another process has an exclusive lock
on the local Symmetrix.
```

To further examine the presence of this lock, use the following form:

```
symcfg -sid SymmID list -lock -lockn ALL
```

### CAUTION

**Use the release action only if you believe the lock was forgotten and there are no other operations in progress to the database.**

The command will list Symmetrix external locks being held. For this case, it will show a number 14 device masking lock and the length of time it has been on.

To release this lock, use the following form:

```
symcfg -sid SymmID -lockn 14 release
```

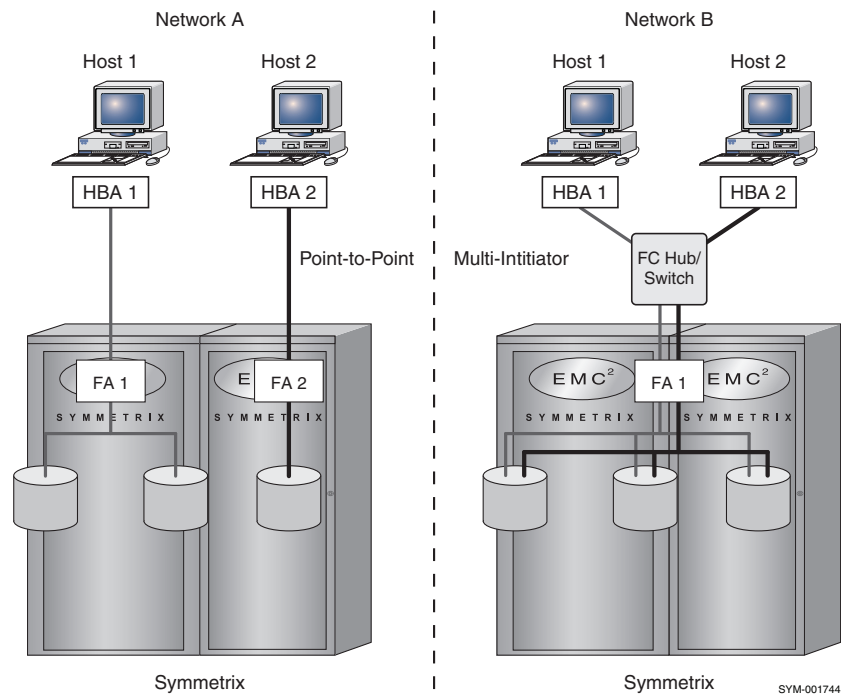
## Supported topologies

Device masking supports Fibre Channel (point-to-point, arbitrated loop, and switch fabric) and native iSCSI topologies.

### Fibre Channel topology

[Figure 32](#) illustrates two different Fibre Channel network configurations that device masking supports:

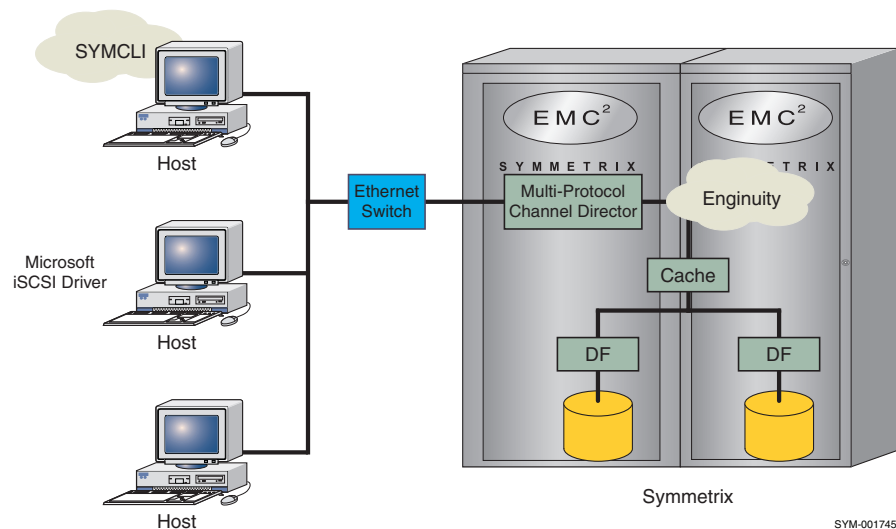
- ◆ **Point-to-point** — Network A (left) is configured with direct connections from one HBA on each host to one FA on the array, providing each host with access to a different set of devices.
- ◆ **Multi-initiator** — Network B (right) is configured with multiple hosts accessing the same Symmetrix devices through a common fabric.



**Figure 32** Point-to-point and multi-initiator topologies

## Native iSCSI topology

In contrast to the Fibre Channel topology, in a native iSCSI environment, hosts are connected to a Symmetrix array through an Ethernet switch, as shown in [Figure 33](#).



**Figure 33** Native iSCSI topology



## CHAP

Native iSCSI support standards require that a security protocol be available. Enginuity version 5670 provides the Challenge Handshake Authentication Protocol (CHAP), which can be enabled or disabled by the user. Refer to [“iSCSI software driver configuration” on page 363](#) for more information about CHAP authentication.

## HBA initiator support

Both HBA and Symmetrix director ports in the topology are uniquely identified by a name (WWN or iSCSI). For ease of use, you can associate an ASCII nickname (AWWN and AISC SI).

SYMCLI device masking supports the following HBA initiators:

- ◆ World Wide Name (*wwn*) and alias for World Wide Name (*awwn*)
- ◆ Native iSCSI over TCP/IP (*iscsi*) and alias for iSCSI over TCP/IP (*aiscsi*)

Refer to [“Discovering host HBAs” on page 332](#) for more information about WWN and iSCSI initiators.

---

**Note:** An iSCSI initiator can log in to the array (through a Gig-E port) before the VCMDB is initialized. After the database is initialized, only iSCSI initiators with a masking record are allowed to log in. If the database is initialized by a host connected to the array through a Gig-E port, a record for the iSCSI initiator of that host will be added to the database so that access to the array is not lost as a result of this command.

---

## Using the device masking database

By default, the device masking database (VCMDB) is accessible to all HBAs that log in to the director port where the database is configured. Thus, any host with access privileges can effectively modify the contents of the database if it has device masking installed.

However, to prevent an unauthorized host from changing the database, the Symmetrix operating system allows you to control a host's access to the database device through the contents of the database records. Only HBAs with valid records in the database can access the database.

The device masking data base (VCMDB) can be unmapped from any director that is not being used for masking control.

---

**Note:** If you have PowerPath installed, you should keep the VCMDB mapped.

---

## Configuring device masking

Before you begin using the Solutions Enabler Device Masking, it is important to understand your device masking and test your scripts in a controlled environment.

When you set up your device masking environment, you should initialize the device masking VCMDB (see [“Initializing the database” on page 338](#)). This should only be done on an initial setup, since it clears the device of any and all data.

Once you understand the Solutions Enabler Device Masking functionality described herein, you can begin to develop device masking scripts customized for your environment.

## Access control environment setup

If Symmetrix Access Control is being used to protect Symmetrix devices, the host from which you run the device masking commands must be configured in an access control group with an ACL (Access Control List) granting VLOGIX rights to ALL\_DEVS. Otherwise, changes to the configuration records in the VCMDB would fail.

Some controls require DIRCTRL or CFGSYM rights. Initialize, restore, and convert controls require CFGSYM and VLOGIX rights. Authentication and `symconnect` controls require DIRCTRL rights.

For more information about the Symmetrix Access Control `symacl` command, see the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

## Configuration steps

Configuring device masking involves four steps:

1. [“Identify configuration components” on page 330](#)
2. [“Initialize and update the database” on page 331](#)
3. [“Enable authentication” on page 332](#)
4. [“Recommendations for activating the configuration” on page 332](#)

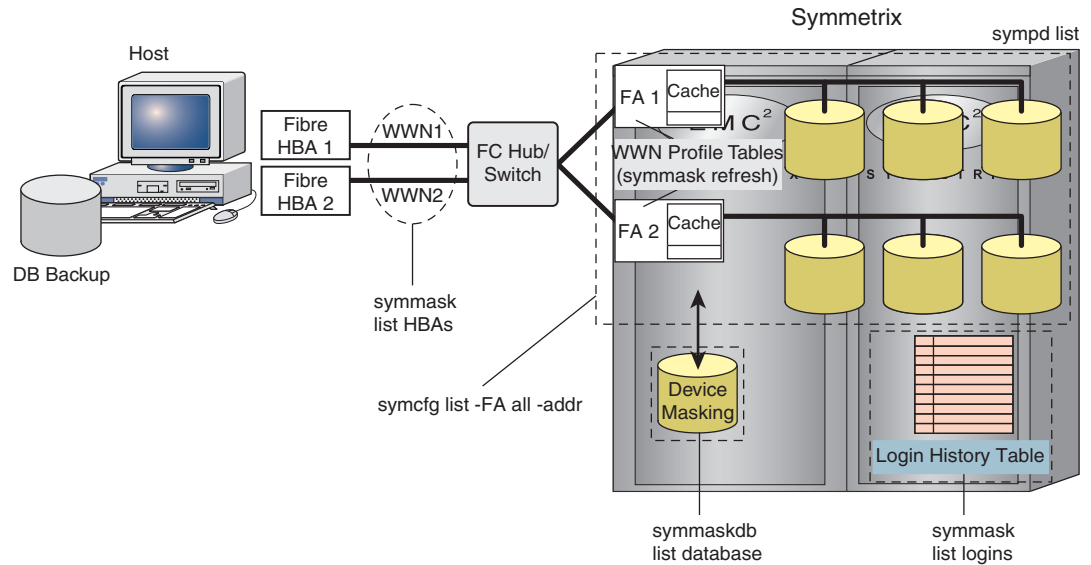
You should be logged on to the control station as Administrator on a Windows system or as root on a UNIX system.

## Identify configuration components

Discover local HBAs on a host that have a channel to a Symmetrix array and generate an AWWN for any HBA that does not have an AWWN assigned yet, updating the VCMDB with the new information.

```
symmask discover hba
```

[Figure 34 on page 331](#) shows device masking components to identify.



SYM-001747

**Figure 34** Device masking components

The following commands can be used to obtain the identifiers of each discovered element:

- ◆ List the Symmetrix physical device names of all the device masking devices:

```
sympd list -vcm
```

- ◆ List the HBAs on a host and the Symmetrix FA director port to which each HBA is connected. Note the WWN of the HBA that you will be configuring and which FA port connects to that HBA:

```
symmask list hba
```

- ◆ Symmetrix director port to which each HBA on the host connects:

```
symmask list logins
```

- ◆ List devices mapped to the FA director that you will be configuring (for example, director 16A):

```
symcfg list -FA 16A -addr
```

SYMCLI device masking supports both World Wide Name (WWN) and native iSCSI (iSCSI) HBA connections.

## Initialize and update the database

After you have identified each element, you can initialize the database and create records. To initialize the database, use the following command:

```
symmaskdb -sid SymmID init -file BackupFilename [-vcmdb_type <3 | 4 | 5>]
```

For more information on initializing the VCMDB, refer to [“Initializing the database” on page 338](#); and for more information on supported database types, refer to [“VCMDB database types” on page 339](#).

Make an entry for the HBA-to-FA connection in the VCMDb, specifying devices that the HBA can access. For example, add a range of devices (0030 through 0034) to the VCMDb on the Symmetrix array (`-sid 814`), specifying the HBA's WWN and the FA director/port that the HBA connects to.

```
symmask -sid 814 -wwn 20000000c920b484 add devs 0030:0034 -dir 16A -p 0
```

## Enable authentication

For detailed information regarding initializing, setting, enabling, and disabling authentication, refer to [Chapter 12, “Device Masking: iSCSI Setup.”](#)

## Recommendations for activating the configuration

To ensure that updates to the VCMDb become active and visible to your host, it is recommended that you:

1. Back up the device masking VCMDb to a file by calling `symmaskdb backup`.

---

**Note:** During the initial setup you cannot create a backup.

---

2. Update the Symmetrix array with the configuration changes by performing `symmask refresh`. This calls the Symmetrix director to refresh its WWN/iSCSI-related profile tables in cache with the contents of the device masking VCMDb.



**Before running the `symmask refresh` command, or the `symmask -verify` command, make sure there are no HBAs accessing devices in the masked channel (applications running or user activity).**

---

3. Reboot all hosts that have had devices added or removed for the changes to take effect.

---

**Note:** When you reboot a host, you must run `symcfg discover` to scan the Symmetrix devices and refresh the SYMAPI configuration database.

---

4. When configuration of the database is complete, use `sympd list` to view the Symmetrix devices that can be seen by the host.

## Discovering host HBAs

During the initial setup, an administrator runs `symmask discover` on the controlling host to search the environment for Symmetrix devices on each HBA by using the following command:

```
symmask discover hba
```

---

**Note:** It is assumed that the host from which these commands are run has access to the Symmetrix devices.

---

When the `symmask discover` finds a host HBA, it reads the login history table and performs the following:

1. Checks whether an alias exists in the device masking VCMDB. If one does, this command writes it to the login history table.
2. If there is no alias in the device masking VCMDB record, or the login history table, it creates an ASCII alias and writes it to the login history table.

---

**Note:** There is a `-rename` option that can be used with this command to force the discovered hostname/HBA name (or IP address) to be written to the login history table and the device masking VCMDB. This will overwrite any existing AWWN/AISCSI record you have previously established.

---

3. Prints the initiator identifier (WWN/iSCSI) of the HBAs that are connected to the masked channel and Symmetrix array.
4. If the `-rename` option was used with the `symmask discover` command, the initiator identifier and its ASCII alias are written to the device masking VCMDB.

The `symmask discover` command sends information about this connection back to its host system. The `discover` command is the primary mechanism by which hosts other than the control station can learn about their VCMDB paths to the Symmetrix array. This is displayed using the `symmask list hba` command.

## Using alias names

Whether you have defined alias names (AWWN/AISCSI) for the various HBAs, or choose to use those assigned by the SYMAPI server during discovery, alias names can be used in the command line, replacing the cumbersome numeric identifiers.

These names, which are stored in the Symmetrix array's login history table, identify the HBAs connected to the network interface. Alias names can be shorter in length and much more recognizable than the cryptic WWNs/iSCSIs.

## ASCII format

All alias names (auto-generated or user-defined) have two parts separated by a slash (/), such as *ALIAS/ALIAS*. An ASCII alias names generated by the `discover` action consists of two parts: the name of the host and the name of the HBA.

- ◆ For Fibre configurations, the adapter number takes the form of the WWN or iSCSI to guarantee uniqueness. For example, the AWWN for a host whose TCP/IP hostname is `john4554b`, on adapter `10000000c920cf87`, would be `john4554b/10000000c920cf87`.
- ◆ For Native iSCSI configurations, the values are hostname/IP address.

For more information on renaming Identifiers, refer to [“Managing HBA initiators” on page 346](#).

## Renaming identifiers

When using various `symmask` actions (such as adding or removing devices in the device mask) you can target an HBA path by specifying an AWWN or AISCSI in the command line. Once you have established the aliases with the identifiers in the history table and database, you can rename existing aliases with `symmask rename` action using the following form:

```
symmask -sid SymmID -iscsi iscsi rename aiscsiNew
```

For example, you are working with Symmetrix 0128 and you want to change your HBA of 20000000c920b484 to Solar2b, enter:

```
symmask -sid 0128 -wwn 20000000c920b484 rename Solar2b/b4
```

You can run `symmask list logins` to display the contents of the login history table to examine the existing alias names on a specified Symmetrix array.

## Adding masked devices

The Symmetrix devices you want to isolate can be assigned to a specified masked channel (HBA to director port). You can add or remove devices from these masked channels using their Symmetrix device name, or device group name.

If an alias is set in the login history table, the first record for the initiator in the database can be added using that alias. In addition, any attribute can be set by alias name if the alias can be found in the login history table.

### Adding devices

To add a device or devices to a specified HBA/director-port channel, use the following form:

```
symmask -sid SymmID -wwn wwn | -awwn awwn |  
        -iscsi iscsi | -aiscsi aiscsi | -host HostName  
        -dir <#|all> -p <#|all> [-celerra] [-rp]  
        [-reserve_id ResvID[, ResvID[, ResvID]]]  
  
add devs SymDevStart:SymDevEnd |  
        <SymDevName | SymDevName, SymDevName, SymDevName...>  
        [-dynamic_lun | -lun Addr | <Addr, Addr...>]  
        [-noprompt] [-remove_unmapped] [-map]
```

For example, to add devices 0014 and 0015 on Symmetrix 0128 for access to all directors and ports on Host3b, enter:

```
symmask -sid 0128 -host Host3b -dir all -p all add devs 0014,0015
```

If the devices are not addressed to the specified FA, a warning message displays.

If the devices are already assigned in the database to any WWN, an informational prompt displays. To turn off this functionality, use the `-noprompt` option.

Use the `-map` option to additionally map the devices while adding them to the masking database.

### Adding metadevices

To add metadevices, add only the *SymDevname* of the device that is designated as the metahead.

### Adding reserved devices

If you are using devices that have been reserved, you must supply the device reservation ID.

For example, to add reserved device 0014 on Symmetrix 0128 for access to Host3b using director 16a, port 0, enter:

```
symmask -sid 0128 -host Host3b -dir 16a -p 0 add dev 0014 -reserve_id 5
```

[“Managing device reservations” on page 78](#) provides more information about device reservations.

## Adding device group devices

To add device group devices to a specified HBA/director-port channel, use the following form:

```
add -g DgName [-std] [-bcv] [-vdev] [-tgt] [-noprompt]  
[-dynamic_lun | -lun Addr] [-remove_unmapped]
```

Use the `-std` option to mask only the standard devices; `-bcv` to mask only the BCV devices; `-vdev` to mask only the virtual devices; and `-tgt` to mask only the target devices.

For example, to add the standard devices of device group `prodB` on Symmetrix 0128 for access to Host3b using director 16a, port 0, enter:

```
symmask -sid 0128 -awwn Host3b/4a add -g prodB -std -dir 16a -p 0
```

If no option is specified, only standard devices (`-std`) will be added.

---

**Note:** Device group names are not stored, only the device information is stored.

---

## Adding devices from a device file

You can perform add masking operations on devices in a device file. When creating the device file, enter the source device first, a space, then the target device. Device files can contain comment lines that begin with the pound sign (#). For example:

```
#Device list mask2 for masking  
0014 009A  
0151 014C  
....
```

A Symmetrix ID is required for this option. In addition you can use the keywords `src` and `tgt` to specify that the action should only use the source devices, or only use the target devices.

To mask only the source devices in the `Mask2` file on Symmetrix 0845, to director FA-2A and port 0, enter:

```
symmask -sid 0845 -wwn 50060482d5f003c1 -dir FA-2A -p 0 add -f mask2 src
```

If the `src` or `tgt` keywords are not specified, only the source devices (`src`) are acted upon.

## Adding devices with dynamic LUN addressing

When adding a list of devices, a device group, or a device range, you can specify the starting LUN address for each device using the `-lun` option, or have SYMAPI assign the LUN address using the `-dynamic_lun` option.

To add a device or devices to a specified HBA/director-port channel, specifying the LUN address use the following form:

```
symmask -sid SymmID -wwn wwn|-awwn awwn|-iscsi iscsi |-aiscsi aiscsi  
  
add devs SymDevStart:SymDevEnd|SymDevName|SymDevName,...\  
-lun Addr | Addr, Addr, Addr...  
-dir # -p # [-noprompt] [-dynamic_lun] [-remove_unmapped] [-map]
```

Where:

**-lun** — Specifies starting LUN addresses. You can specify a single starting LUN or multiple LUNs to match the given ranges. For example:

```
symmask add devs 15,18,20 -lun 0 -wwn 20000000c920b484 -dir 14C -p 1 -sid 12
```

**-dynamic\_lun** — Specifies the use of dynamic LUN addressing, however the application assigns the addresses based on what is already in use for the host HBA.

**-removed\_unmapped** — Allows you to remove any unmapped devices when adding devices with dynamic addressing. This option converts an existing device masking record to the new dynamic feature in a single step. Records that contain unmapped devices cannot be converted until those devices are removed.

**-map** — Allows you to map devices while adding them to the masking database. This option takes the next available LUN, which may or may not match the masking LUN. The **-map** option is only supported on Enginuity 5773.

*Examples* To add devices 100 to 105 on Symmetrix 104 for access to wwn 20000000c920b484 starting at LUN addresses 0, and using the next available LUN numbers, for director 2d, port 0, enter:

```
symmask add devs 100:105 -lun 0 -wwn 20000000c920b484 -dir 2d -p 0 -sid 104
```

When adding devices from a device file or device group, add the **-dynamic\_lun** option to the existing syntax for the system to assign the LUN addresses. For example:

```
symmask add -f C:\my_device_file -dynamic_lun -wwn 20000000c920b484 -dir 3c -p 0 -sid 90
```

```
symmask add devs 2C,2E,30 -dynamic_lun -wwn 20000000c920b484 -dir 2a -p 1 -sid 215
```

## Removing masked devices

You can remove some of the devices associated with an HBA entry, or you can remove the entire set of devices associated with an HBA entry. Removing some devices (but not all) requires syntax similar to that for adding devices and the same refresh, backup, and discover steps after completing the remove operation. You can remove devices from a masked channel at any time. To remove a device or devices from a masked channel, use the following form:

```
symmask -sid SymmID -wwn wwn|-awwn awwn|-iscsi iscsi|-aiscsi aiscsi  
-name <name> | -host HostName -dir <#|all> -p <#|all>  
[-celerra] [-reserve_id ResvID [,ResvID [,ResvID]]]
```

```
remove devs SymDevStart:SymDevEnd | SymDevName |  
SymDevName, SymDevName, SymDevName... [-unmap] [-force]
```

The **-unmap** option allows you to unmap the devices while removing them from the masking database, if the devices are not masked to any other initiator on that port. This option is only supported for Enginuity 5773.150.

The **-force** option may be useful when you want to quickly remove a range of specified device names that might span device names not part of the existing noncontiguous masked channel. It may also be needed when working with metadevices.

After each set of changes, activate the configuration by performing a refresh (**symmask refresh**), back up the database (**symmaskdb backup**), and reboot the affected hosts.



---

**Note:** When ever you reboot a host, you must run `symcfg discover` on all the Symmetrix devices and refresh the SYMAPI database.

---

*Examples* To remove devices 0031 and 0033 from the 0030-to-0034 range of devices that was added previously:

```
symmask -sid 814 -wwn 20000000c920b484 remove devs 0031,0033 -dir 16A -p 0
```

To remove the remaining devices in the 0030-to-0034 device range, you can specify individual devices or the range with an option (`-force`) that allows you to remove a noncontiguous range. For example:

```
symmask -sid 814 -wwn 20000000c920b484 remove devs 0030:0034 -dir 16A -p 0 -force
```

To remove the entire set of devices that an HBA can access, use `symmask delete` and specify the WWN of the HBA. The delete action removes the HBA entry completely, including any attributes set previously. For example:

```
symmask -sid 814 delete -wwn 20000000c920b484
```

## Removing metamembers

To remove metamembers from the device masking VCMDDB, but keep the metaheads in place, use the following form:

```
symmaskdb -sid SymmID -meta_member remove
```

The `-force` option may be necessary when working with metamembers.

## Removing device group devices

To remove device group devices from a specified HBA/director-port channel, use the following form:

```
symmask -sid SymmID -wwn wwn|-awwn awwn|-iscsi iscsi |-aiscsi aiscsi  
remove -g DgName [-std] [-bcv] [-vdev] [-force] -dir # -p # [-noprompt]
```

Use the `-std` option to remove only the standard devices; `-bcv` to remove only the BCV devices; and `-vdev` to remove only the virtual devices. For example, to remove the standard devices of device group `prodB` on Symmetrix 0128 from access to Host3b using director 16a, port 0, enter:

```
symmask -sid 0128 -awwn Host3b/4a remove -g prodB -std -dir 16a -p 0
```

If no option is specified, only standard devices (`-std`) will be removed.

---

**Note:** Device group names are not stored, only the device information is stored.

---

## Removing devices using a device file

To remove the masking entries for only the source devices in the `Mask2` file on Symmetrix 0845, to director FA-2A and port 0, enter:

```
symmask -sid 0845 -wwn 50060482d5f003c1 -dir FA-2A -p 0 remove -f mask2 src
```

If the `src` or `tgt` keywords are not specified, only the source devices are acted upon.

## Maintaining the device masking database

After the initial setup, records are added, updated, and deleted from the device masking VCMDb each time a SYMCLI command is issued to add or remove devices in a masked channel, a WWN /iSCSI alias is renamed, one HBA is swapped for another, or an HBA is cleared, removing its device mask allocation.

### Initializing the database

For the initial setup of any device masking environment, a device reserved (VCM state enabled) in the Symmetrix array must be initialized and formatted for use as the device masking database, better known as the VCMDb.

The initialization clears the disk device of any current data in the process of formatting the database. In most cases, you do not want to clear the data of an existing VCMDb. If you are unsure whether a VCMDb currently exists, issue the command that displays an existing VCMDb. For example, to view the VCMDb on Symmetrix 0128:

```
symmaskdb -sid 0128 list database
```



**The following command is rarely used. Be sure you want to zero out the device masking VCMDb before proceeding.**

To initialize and clear the database device, use the following form:

```
symmaskdb -sid SymmID init -file BackupFilename [-vcmdb_type <3 | 4 | 5>]
```

For a safeguard, you must specify a backup filename, since this command will try to write the data from this device to a backup file on your host before it clears the current data.

For example, to initialize the database and create backup file `BackupDevMask1` on Symmetrix 0128, enter:

```
symmaskdb -sid 0128 init -file BackupDevMask1
```

If you do not include the `-vcmdb_type` option, the default database type depends on the size of the VCMDb device.

### Preventing unauthorized modification of the VCMDb

By default, the VCMDb grants access to all HBAs that log in to the FA director port where the database is configured. Without preventive measures, any host with access privileges can modify the VCMDb if it has the EMC Solutions Enabler Device Masking component.

One way to prevent unauthorized host access is by using the Configuration Manager to set the `VCMDb_restricted_access` Symmetrix parameter. This metric is obsolete in Enginuity version 5771 and higher.

### Managing a backup VCMDb file

You can create a backup file containing the current contents of the device masking VCMDb. This is useful when you want to temporarily change the access rights or device masking assignments to various HBAs. Then at some point in time, you can return the

device masking environment back to the original masked environment. Often, just backing up the database on a regular basis ensures you can recover your established masked environment in the event of some improper changes or failure.

---

**Note:** You cannot reuse any existing backup filename. Always create a new file.

---

To create a backup database file, use the following form:

```
symmaskdb -sid SymmID backup -file BackupFilename
```

For example, to create backup file BackupDevMask on Symmetrix 0128, enter:

```
symmaskdb -sid 0128 backup -file BackupDevMask
```

The VCMDB backup files vary in length, depending on their contents.

## VCMDB database types

Device masking database types define the number of devices per database record and the number of records you can mask.

Solutions Enabler version 5.3 and Enginuity version 5670 introduced support for a Type 4 device masking VCMDB, which expanded the existing Type 3 VCMDB option.

Beginning with Solutions Enabler version 6.0 and Enginuity version 5671, a Type 5 device masking VCMDB is supported, and Enginuity version 5771 will only support a Type 6 VCMDB. The various VCMDB types are defined as:

- ◆ **Type 3** — Supports up to 8 K devices per record and 32 fibre/32 iSCSI initiator records per port (the VCMDB device must be 24 cylinders or larger)
- ◆ **Type 4** — Supports up to 8 K devices per record and 64 fibre/128 iSCSI initiator records per port (the VCMDB device must be 48 cylinders or larger)
- ◆ **Type 5** — Supports up to 16 K devices per record and 64 fibre/128 iSCSI initiator records per port (the VCMDB device must be 96 cylinders or larger)
- ◆ **Type 6** — Supports up to 64 K devices per record and 256 fibre/512 iSCSI initiator records per port (cannot be specified with SYMCLI)

When initializing the VCMDB database where no database currently exists, SYMCLI defaults to creating a database according the size of the VCMDB device being initialized. For example, a Type 4 VCMDB would be created for a 48-cylinder device. This default behavior is also true for the VCMDB device that currently holds a database (for example, a Type 4 database on a 96-cylinder device will be initialized as a Type 5 database). A Type 4 or 5 database initializes with direct I/O writes blocked to protect against outside sources corrupting the database. You can also block direct I/O writes to a Type 3 database using `set no_direct_io` option.

You can explicitly create a VCMDB of a specific type in the initialization procedure. This is especially important when you are considering backwards compatibility issues. If you had a larger VCMDB device size (96 cylinders, for example), but you have a need to be compatible with Solutions Enabler version 5.3, you would probably want to specify the smaller size (Type 3 or 4, for example) to maintain backward compatibility. The following command provides an example syntax specifying a VCMDB type:

```
symmaskdb -sid 814 init -file MyInitBackup -vcmdb_type 4
```

## Converting a VCMDb type

You can convert a Type 3 database to Type 4 or Type 5, and a Type 4 to a Type 5, if the size of the VCMDb device on the specified Symmetrix array is large enough for the database type. The following is an example converts the VCMDb on Symmetrix 814 to a Type 5 database:

```
symmaskdb -sid 814 convert -vcmdb_type 5 -file MyCvrtBU
```

If you convert from a lower type database to a higher type, any hosts running a Solutions Enabler version that does not support the higher VCMDb type will not be able to access the database. For example, if one host running Solutions Enabler version 6.0 converts an existing VCMDb to Type 5, another connected host running Solutions Enabler version 5.4 will no longer have access to the database until this host is upgraded to version 6.0 or higher.

## Restoring a backup VCMDb

You can restore the database from the backup file stored on the host, as follows:

```
symmaskdb -sid 0128 restore -file MyCvrtBU
```

The database is restored as is; Type 3 restores to Type 3, and Type 4 restores to Type 4. You can use the `convert` or `set` options to alter the resulting environments.

To restore the database from a backup file, but not the authentication information, enter:

```
symmaskdb -sid 0128 restore -file MyCvrtBU -skip_authentication
```

---

**Note:** The `-skip_authentication` option is not available with Enginuity 5771 and higher.

---

## Restoring a backup and converting its type

To restore from a backup file and convert its database type in a single command specify the `-vcm_type` option and specify the type: 4 or 5. For example, enter:

```
symmaskdb -sid 0128 restore -file MyCvrtBU -vcm_type 5
```

## Blocking direct writes to the VCMDb

If you have a Type 3 database and would like to block direct writes to the database, use the following form:

```
symmaskdb -sid SymmID set -no_direct_io
```

---

**Note:** Type 4 and 5 databases initialize with direct I/O writes blocked by default.

---

# Viewing the VCMDb

Once you have initialized your database and created records, you can view various information about the database. In addition, you can view the login history details.

## Viewing the database

You can examine the entire contents of the device masking VCMDb in its entirety, by director and port, by WWN name (or alias), or by iSCSI name (or alias) using the following form:

```

symmaskdb -sid SymmID | -file Filename [-v]
[-dir all [-p all] | -dir # [-p <#|all>]]
[-wwn wwn | -awwn awwn |
-iscsi iscsi | -aiscsi aiscsi]
list database

```

## Example: Entire VC MDB

To view the device masking VC MDB on Symmetrix 6196, enter:

```
symmaskdb -sid 6196 list database
```

The following is sample output from this command:

```

Symmetrix ID           : 000000006196

Database Type          : Type4
Last updated at        : 04:58:00 PM on Tue Mar 25,2004

Director Identification : FA-2A
Director Port          : 1

User-generated
Identifier             Type   Node Name      Port Name      Devices
-----
10000000c9238053      Fibre  api145         i@1f,4000,@2   0040:0043
10000000c924e04a      Fibre  HOST.23.65.70  10000000c924e04a 00BC:00BF
                                           00C3:00C6

Director Identification : FA-2B
Director Port          : 1

Identifier             Type   Node Name      Port Name      Devices
-----
10000000c9238053      Fibre  api145         i@1f,4000,@2   None

```

## Example: iSCSI connected type 4 database

The following is sample output for a Type 4 database connected through iSCSI:

```

Symmetrix ID           : 000000006208

Database Type          : Type4
Last updated at        : 03:29:45 PM on Fri Jul 25,2004

Director Identification : SE-3A
Director Port          : 0

Identifier             Type   Node Name      Port Name      Devices
-----
iqn.2002-06.com*      iSCSI  iSCSI          microsoft:api210 0001
                                           0005:0007
                                           0047
                                           004F

Director Identification : FA-14A
Director Port          : 0

Identifier             Type   Node Name      Port Name      Devices
-----
2234567812345678      Fibre  2234567812345678 2234567812345678 0060
1234567812345678      Fibre  1234567812345678 1234567812345678 0060

```

## Example: VCMDB records for a director

To examine the database for records concerning director 2b, port 1 on Symmetrix 6196, enter:

```
symmaskdb -sid 6196 list database -dir 2b -p 1
```

The following is sample output from this command:

```
Symmetrix ID           : 000000006196
Last updated at        : 04:58:00 PM on Tue Mar 25,2004
Director Identification : FA-2B
Director Port          : 1
```

Identifier	Type	User-generated		Devices
		Node Name	Port Name	
10000000c9238053	Fibre	api145	i@1f,4000,@2	None

You can examine the masked assignment of devices to a specific HBA using the following form:

```
symmaskdb -sid SymmID list devs [-wwn wwn | -awwn awwn|-iscsi iscsi |
-aiscsi aiscsi]
```

**Note:** The initiator is newly optional with Solutions Enabler 7.1.

To examine the devices on Symmetrix 343 without supplying a specific initiator, enter:

```
symmaskdb -sid 343 list devs
```

The following is sample output from this command:

```
Symmetrix ID           : 000190300343
Originator Port wwn    : 210000e08b04daac
User-generated Name    : HOST.23.193.182/210000e08b04daac
```

Sym Dev Name	Dir:P	Physical Device Name	VBUS	TID	LUN		Attr	Cap (MB)
					SYMM	HOST		
03A7	2C:0	N/A	-	-	-	1	-	-

```
iSCSI Name:
iqn.1991-05.com.microsoft:tk3-cu-hv34.parttest.extranetest.microsoft.com
User-generated Name : api82/172.23.199.16
```

Sym Dev Name	Dir:P	Physical Device Name	VBUS	TID	LUN		Attr	Cap (MB)
					SYMM	HOST		
0275	15C:0	Not Visible	0	0	1	N/A	754	
0572	15C:0	Not Visible	0	0	f00	N/A	188	

```
iSCSI Name: iqn.1991-05.com.microsoft:usensthammal4e.eng.emc.com
User-generated Name : /
```

No Devices were found for that iSCSI Name

**Note:** The appropriate physical device name now appears wherever multiple path information for HP-UX Agile devices is provided.

## Viewing device capacity

You can view the capacity of devices assigned to a particular host with the following command:

```
symmaskdb -sid SymmID list capacity -host HostName
```

When a hostname is specified, such as `api145`, SYMAPI looks for records with the AWWN value of `api145/` and pull that record for the device information. If the default AWWN alias has been modified to something other than the host name, this command does not include information for that HBA, or may appear incomplete if only 2 out of 3 HBAs for a given host have the hostname in the alias.

**Note:** The `list database` and `list devs` commands can be targeted to a backup database file on your host by replacing the `-sid` option with a `-file` option that specifies your backup filename.

For example, to view the capacity of host `api145` on Symmetrix 6196, enter:

```
symmaskdb -sid 6196 list capacity -host api145
```

The following is sample output from this command:

```
Symmetrix ID      : 000000006196

Host Name         : api145
Identifiers Found : 10000000c9238053

Device  Cap(MB)  Attr  Dir:P
-----  -
0040      187      2A:1
0041      187      2A:1
0042      187      2A:1
0043      187      2A:1

-----

MB Total:      748
GB Total:      0.7
```

This command requires that the first part of the HBA alias be the host name.

## Viewing HBA assignments

You can view which HBAs have been assigned to specific devices with the following command:

```
symmaskdb -sid 6196 list assignment -devs 0040:0043
```

The following is sample output from this command:

```
Symmetrix ID : 000000006196

Device  Identifier          Type  Dir:P
-----  -
0040     10000000c9238053  FIBRE  2A:1
0041     10000000c9238053  FIBRE  2A:1
0042     10000000c9238053  FIBRE  2A:1
0043     10000000c9238053  FIBRE  2A:1
```

Use the `list no_assignment` command to list devices that are mapped, but not yet assigned, in the device masking VCMDDB.

## Viewing the login history table

The `symmask list logins` command is used to view the login history table. This table in the Symmetrix array lists which hosts and HBAs are logged on to a Symmetrix array for all directors and their director ports. You can also use options to restrict this data to a specific director and/or port number.

For example to return login information for Symmetrix 6196, enter:

```
symmask -sid 6196 list logins
```

The following is sample output from this command:

```
Symmetrix ID           : 000000006196

Director Identification : FA-2A
Director Port          : 1
```

Identifier	Type	User-generated		FCID	Logged On	
		Node Name	Port Name		In	Fabric
10000000c9238053	Fibre	api145	i@1f,4000,@2	260e13	Yes	Yes
5006048000060d21	Fibre	NULL	NULL	261e13	No	Yes

The identifier field indicates which HBA is communicating with the Symmetrix array. User-generated node and port names are identified as the AWWN or AISCSI alias associated with it. Columns labelled On Fabric and Logged In indicate whether the HBA is connected to a fabric and whether it is logged in to the Symmetrix system.

You can use the verbose (`-v`) option to view the last active login information.

## Refreshing the database

The device masking records are located in two places on the Symmetrix array: the database and the local directors. When a record is written to the array, it is stored in the database and then pushed out to the directors at a later time. A `symmask refresh` command will trigger all records within the database to be pushed out to the local directors. A login by an HBA only triggers the record for that specific HBA to be updated.

To view what is currently stored on the local directors to see what (if any) affect a `symmask refresh` command will have, use the `-verify` option, as shown in the following example:

```
symmask refresh -verify -sid 0237
```

The example output follows:

```
Symmetrix ID           : 000190300237

Database Type          : Type6
Last updated at        : 12:27:58 PM on Thu Sep 13, 2007

Director Identification : FA-16C
Director Port          : 0
VCM Enabled            : No

    Originator Port wwn  : 5006048ad5f004ce
    Type                 : Fibre
- Record Location       : Director only
- Devices               : 0011:001E,0037:0040,0048:004B,0C1B:0C20,
                        0C28:0C38,0C96

Director Identification : SE-2D
```



```

Director Port          : 0
VCM Enabled            : No

    iSCSI Name: iqn.1991-05.com.microsoft:user.eng.emc.com
    Type            : iSCSI
    Record Location   : Both, Database and Director
    Devices          : Do Not Match
-    Director Devices : 0033:0036

    iSCSI Name: iqn.1991-05.com.microsoft:user.eng.emc.com
    Type            : iSCSI
-    Record Location   : Director only
-    Devices          : 0033:0036

    iSCSI Name: iqn.1991-05.com.microsoft:user.eng.emc.com
    Type            : iSCSI
-    Record Location   : Director only
-    Devices          : 0033:0036

    iSCSI Name: iqn.1991-05.com.microsoft:user.corp.emc.com
    Type            : iSCSI
-    Record Location   : Director only
-    Devices          : 0033:0036

    iSCSI Name: iqn.1991-05.com.microsoft:user.eng.emc.com
    Type            : iSCSI
-    Record Location   : Director only
-    Devices          : 0033:0036

    iSCSI Name: iqn.1991-05.com.microsoft:api188
    Type            : iSCSI
-    Record Location   : Director only
-    Devices          : 0033:0036

Director Identification : FA-16D
Director Port          : 0
VCM Enabled            : No

    Originator Port wwn : 10000000c9239664
    Type                : Fibre
    Record Location      : Both, Database and Director
    FCID Lockdown        : On
    FCID Values          : Do Not Match
+    Database FCID       : 123456
-    Director FCID       : 624e13
    Port Flag Overrides  : On
+    Database Enabled     : Disable_Q_Reset_on_UA(D)
                           Environ_Set(E)
    Disabled Flags       : Match
    Devices              : Do Not Match
    Matching Devices     : 001A:001C
-    Director Devices    : 0017:0019,0037:0041,0C1B:0C2E,0C43,0C95,
                           0C99:0CB6,0CC1:0CE8

    Originator Port wwn : 210000e08b04daac
    Type                : Fibre
    Record Location      : Both, Database and Director
    FCID Lockdown        : On
    FCID Values          : Match
    Lun Offset           : On
    Offset Value         : Do Not Match
+    Database Offset     : 0100
-    Director Offset     : Not Set
    Base Value           : Do Not Match
+    Database Base       : 0020
-    Director Base       : Not Set

```

```

Devices          : Do Not Match
  Matching Devices : 0017:0019
-   Director Devices : 001A:001C, 0037:0041, 0C1B:0C2E, 0C43, 0C95,
                        0C99:0CB6, 0CC1:0CE8

```

Legend: + = Added from database, - = Removed from director

This output is similar to the `list database -v` output, with the following differences:

- ◆ Includes the state of the VCM flag on the port.
- ◆ The first column contains either a minus (-) or a plus (+) when information within the records do not match, or when the record is located in only one location and will be completely added or removed.
- ◆ Attributes, such as LUN offset, which are not set in the record(s) are not displayed.

## Managing HBA initiators

An HBA can be specified in `symmask` commands by its unique WWN, by an AWWN alias associated with the WWN, or by an iSCSI name (or its alias). These names appear in the Symmetrix login history table and in the VCMDB.

---

**Note:** Fiber HBAs have a host WWN and a port WWN for each port on the HBA. Most HBAs have only one port, but some have two. In rare cases, an HBA has four ports. HBA identification in device masking always refers to a port WWN.

---

When you issue the `symmask discover hba` command to update the login history table, the Symmetrix API (SYMAPI) checks the VCMDB to determine whether an AWWN exists for each WWN record. If not, SYMAPI creates an AWWN that consists of two parts (the name of the host and the name of the HBA) and writes it to the login history table. However, you can rename the AWWN to a shorter name, if you prefer.

The following command allows you to examine AWWNs in the login history table of Symmetrix 814:

```
symmask -sid 814 list logins
```

### Creating and renaming aliases

To assign an AWWN to fit your naming requirements, you can use the `symmask rename` command at any time—even before SYMAPI generates an AWWN. For example, the following command assigns the two-part name `Solaris3A/b4` as the AWWN for WWN `20000000c920b484`:

```
symmask -sid 814 -wwn 20000000c920b484 rename Solaris3A/b4
```

If later you decide you want to replace all user-defined AWWNs (like `Solaris3A/b4`) with system-generated AWWNs, the following command overwrites all existing AWWN entries in the VCMDB:

```
symmask discover hba -rename
```

You can identify the HBA in `symmask` commands through an iSCSI name. The iSCSI is a SCSI-over-IP protocol that tries to take advantage of the IP network for SCSI traffic rather than requiring a new Fibre network. An iSCSI name is used like an IP address or a WWN

and is displayed using the `symmask list logins` or the `symmask list hba` commands. A unique iSCSI name is determined by the hardware that logs into a Symmetrix array and, like the AWWN, a two-part alias name is generated from the `symmask discover hba` command.

## Swapping a new HBA for a failed HBA

In the event a host adapter fails, or needs replacement for any reason, you can replace the adapter and assign its set of devices to a new adapter by using the `replace` action in the following form:

```
symmask -sid SymmID -wwn wwn|-awwn awwn|-iscsi iscsi|-aiscsi aiscsi
replace wwnNew | iscsiNew
```

To swap HBAs, it is suggested to:

1. Run `symmask list logins` to view the old WWN/iSCSI HBAs.
2. Swap HBA boards.
3. Run `symmask list hba` or `discover` to view the new initiator (for example WWN).
4. Run `symmask replace` to substitute a new WWN for all occurrences in the database of the old WWN. For example, to replace old WWN 20000000c920b484 with new WWN 20000000c920b393:

```
symmask -sid 814 -wwn 20000000c920b484 replace 20000000c920b393
```

5. Run `symmask discover -rename` or `symmask rename` to establish the new AWWN and assign an AWWN to the new HBA in both the VCMDDB and the login history table.
6. Run `symmask refresh` to update the director profile tables (in cache) from the database.

## Deleting HBA associations

You can also delete (in the database) the set of devices associated to a host adapter by using the `symmask delete` action with the following form:

```
symmask -sid SymmID delete -wwn wwn|-awwn awwn|-iscsi iscsi|-aiscsi aiscsi [-login]
```

For this database record deletion, you can restrict the action to just devices on a specific Symmetrix director and port with the following option:

```
[-dir #|all -p #|all]
```

In addition, on arrays running Enginuity 5671 and higher, you can use the `-login` option to delete the entry from the login history table as well.

To delete a set of devices associated to a host adapter, it is suggested to:

1. Run `symmask delete` action.
2. Update the Symmetrix array with the configuration change by performing a `symmask refresh`. This calls the Symmetrix director to refresh its WWN/iSCSI-related profile tables in cache with the contents of the device masking VCMDDB.

**⚠ CAUTION**

Before running the `symmask refresh` or the `symmask refresh -verify` commands, make sure there are no HBAs accessing devices in the masked channel (applications running or user activity).

## Managing the Fibre Channel-to-host interface

Using the device masking commands, you can adjust the protocol characteristics of the Fibre Channel-to-host interface to be compatible with your host platform-specific requirements.

For your specific host communication protocol, the `symmask set` command allows an *advanced* user to adjust the following attributes on a host adapter port basis, or all ports by using `-dir ALL / -p ALL`:

- ◆ Fibre Channel ID (FCID) lock down
- ◆ Device LUN visibility
- ◆ LUN base/offset skip
- ◆ Heterogeneous host configuration

A record for the host adapter port assignment must already exist in the VCMDB for these channel attributes to be set.

**⚠ CAUTION**

Do not proceed with any of these adjustments unless you are comfortable with your understanding of the details of your HBA interfaces. Improper settings can disable the use of your host with the Symmetrix array.

### Locking down a Fibre Channel ID

Fibre Channel ID (FCID) lockdown is a security feature that limits host device access by adding Fibre Channel ID information of a switch within a fabric to device access records in the device masking VCMDB. This feature handles WWN spoofing and the threat it poses to your networked systems in a shared (same director port) storage port configuration.

For example, to implement the Fibre Channel ID lockdown feature on Fibre Channel 021300 for director 16A, port 0, enter:

```
symmask -sid 018 set lockdown on 021300 -awwn SolarB/1f,0,fca@1,0 -dir 16A -p 0
```

This feature lets you set the Fibre Channel ID (FCID) of the WWN of the HBA you want to protect. The FCID is then added to the database record for the WWN of the specified HBA with the specified director and is locked. Once a Fibre Channel ID is locked, no user with a spoofed WWN can log in. If a user with a spoofed WWN is already logged in, that user loses all access through that HBA.

**⚠ CAUTION**

When an HBA logs into a director port, the Fibre Channel ID accompanies it, telling the director port where to send its response. By specifying Fibre Channel ID information of the switch (in addition to the WWN of the HBA in the device masking record), the valid

physical path through the SAN for a particular HBA is locked down. Only an HBA with a Fibre Channel ID that matches the FCID specified in the device masking record is able to log in to the storage port. If the incorrect Fibre Channel ID is added to the device masking VCMDb, that HBA will lose access and the host utilities may hang on the server with the locked out WWN. It is recommended that at least two HBAs be available on the administrator host. If one HBA becomes locked out, the host will have access through the other HBA and can correct the record in the database.

---

## Lockdown steps

To find the Fibre Channel ID, lock it down, verify that it is locked down, and then force the change to take effect, use the following procedure:

1. Find the WWN. If the device for the device masking VCMDb is visible, run `symmask list hba` to find the device path of the HBA you want to protect.

---

**Note:** If the VCMDb is unmapped, no PDEVs will be visible when the `symmask list hba` command is issued. The `sympd` command must be called to set an alternate path.

---

2. Find the Fibre Channel ID value by using one of the following methods:
  - Run `symmask list logins -pdev`, specifying the device path you found in step 1, to find the Fibre Channel ID of the WWN of the HBA you want to protect.
  - Find the Fibre Channel ID value on the switch, refer to [“Finding the FCID of a switch” on page 350](#).
3. Run `symmask set lockdown` set to on with the FCID of the Fibre Channel ID you found in step 2.
4. Run `symmaskdb list database` in verbose mode (`-v`) to verify that the Fibre Channel ID is locked down.
5. Either reboot the host or pull the cable from the director and then replace the cable. This causes the change to take effect. If you reboot, you must run `symcfg discover` to refresh the SYMAPI database.

## Effects on other commands

This section describes how locking down a Fibre Channel ID affects other commands:

- ◆ `symmask delete` — Locking down a Fibre Channel ID has no effect on the `delete` action. The specified record is completely cleared from the database.
- ◆ `symmask replace` — Locking down a Fibre Channel ID has no effect on the `replace` action when the cable is simply moved from one HBA to another and not moved at the switch. In this case, the Fibre Channel ID value that is already in place in the database remains the same for the new HBA.

However, if the cable is moved from one port on the switch to another, the FCID value changes. Do not unlock the Fibre Channel ID during this swap. Instead, leave at least one path open to the database device, and reset the FCID value after the swap by recalling the `set` action.

Since you do not have a path from the HBA whose Fibre Channel ID you want to lock down, you cannot use `symmask list logins` to find the FCID value. Instead you must obtain the FCID value from the switch.

## Finding the FCID of a switch

This section describes how to find the Fibre Channel ID on Connectrix® and Brocade switches:

- ◆ Connectrix switch — Through the hardware view, click the board and then the port of the switch whose Fibre Channel ID you want to find. Right-click to display the port properties window that includes the FCID value.
- ◆ Brocade switch — Telnet to the switch and run `nsShow`. Look for the PID value of the WWN of the HBA you want to protect, which is the Fibre Channel ID value.

## Format of a FCID

The Fibre Channel ID basically incorporates the port and the domain ID of the switch in the fabric into which the HBA is plugged.

Connectrix ED-1032 and Brocade 1000 series:

2 2 **04** 1 3

Underlined text is the domain. ***Bold Italic text*** is the port.

In this example, the domain is 2 and the port is 04.

---

**Note:** For Connectrix, the port is offset by 4.

---

Brocade 2000 series DS-16B:

0 2 1 **3** 0 0

Underlined text is the domain. ***Bold Italic text*** is the port.

In this example, the domain is 02 and the port is 3.

## Setting device LUN visibility

The device LUN visibility feature allows the host driver to discover devices with noncontiguous LUN addresses. During the process of discovery, the host operating system scans for LUNs starting at 000 and continuing to a point where it does not find a LUN in the sequence. If there is no LUN 000 on the target director, or there is a break in the sequence of LUNs on that target, some operating systems (notably HP-UX and Linux) do not detect the remaining LUNs and fail to discover noncontiguous devices.

To allow your host to detect all these devices, turn on the device LUN visibility feature so that all devices attached to a specified FA director and port are made available to the HBA. For example, to turn on visibility for FA director 16A, port 0, when working with a host HBA whose WWN is 20000000c920b484:

```
symmask -sid 814 set visibility on -dir 16A -p 0 -wwn 20000000c920b484
```

Allowing the host to scan the other devices past a break in the LUN sequence does not change how the host accesses them. For example, if an HP-UX host has LUNs 0000, 0002, 0003, 0004 assigned to it, that is exactly what the host will see. If you need the host to

access this same sequence without the break between 0000 and 0002, you can adjust host visibility by offsetting LUN addresses as described in [“Setting the LUN base/offset skip adjustment” on page 351](#).

## Setting the LUN base/offset skip adjustment

Certain host platforms require LUN 000 to be present when it scans the interface for devices. For this feature, LUN refers to the SCSI LUN number. Also, these host types and others cannot see devices beyond the initial contiguous LUN sequence (they cannot skip over masked holes in an array of intended devices). In a device masking environment, this can be a problem when you need to mask out certain devices from the visibility range of certain host platforms.

For these host platforms, the device masking LUN base/offset skip adjustment feature provides the ability to specify a LUN *base* and an *offset* hexadecimal value for the skip hole (recorded in the database). When the host asks for a LUN that is equal to, or greater than the skip hole base value, the offset is added to the LUN value requested by the host to render the actual LUN (device) in the Symmetrix array. The *base* value is essentially the host's first missing LUN in the skip hole. The offset is the hole size (number of addresses needed to skip over the hole).

To set LUN base and offset values for a skip hole within an HBA to director channel, use the following form:

```
symmask -sid SymmID set lunoffset on offset base -awwn awwn -dir # -p #
```

For example, (with director 16A/port 0) to make LUNs (devices) 005 through 008 available to host HPB03/1, you need a LUN base address of 000 and an offset of 5 (to skip over 000-004):

```
symmask -sid 018 set lunoffset on 5 0 -awwn HPB03/1 -dir 16A -p 0
```

## Multiple hosts and broken sequences

On the other hand, if your masked devices for an HBA-to-FA connection had LUN addresses 0000–0003 and 0007–0009, you would need to specify a LUN base address of 0004 and an offset of 0003 (to renumber LUNs 0007–0009 as LUNs 0004–0006). You can record only one gap per HBA-to-FA connection.

If you have multiple hosts that cannot discover devices with noncontiguous LUN addresses, you need to issue the `symmask set lunoffset` command for each host.

As shown in [Table 28](#) for Scenario 1, you could have LUN devices 000 through 006 assigned to Host A and 007 through 00A assigned to Host B. Because in this case, Host B needs to see LUN 000 first, you would have to `set lunoffset on` with a base value of 000 and an offset of 7. For this case, Host A does not have a problem since there is no hole in its assigned device sequence and it starts with 000.

**Table 28** LUN base/offset scenarios for multiple hosts with skip holes

	Host A LUNs	Host B LUNs	base	offset
Scenario 1	000-006		-	-
		007-00A	000	7

**Table 28** LUN base/offset scenarios for multiple hosts with skip holes

	Host A LUNs	Host B LUNs	base	offset
Scenario 2	000-002		-	-
	007-008		003	4
		003-006	000	3

For Scenario 2, you could have LUNs 000 through 002 and 007 through 008 assigned to Host A. Host B could have LUNs 003 through 006. Host A's 000-002 is not a problem, but LUNs 007-008 require a skip hole base value of 003 (because the first visible sequence stopped at 002) and an offset of 4 (hole size). Also, Host B's LUNs 003-006 requires a skip base value of 000 and an offset of 3. This scenario would require two commands: one targeting Host A and one targeting Host B. Only one skip hole per HBA channel can be recorded in the database.

## Setting the HBA port flags

HBA port flags can be set on a per initiator basis or by host name. This feature allows specific host flags to be enabled and disabled on a specified director and port or on all directors and all ports. Setting a port flag by host name or for all ports requires Enginuity 5773.150 and higher.

**Note:** Setting HBA port flags replaces setting the heterogeneous host configuration flags in the next section. The heterogeneous host configuration types listed in [Table 29 on page 354](#) continue to be valid, but will not be expanded. To switch to setting HBA port flags, the heterogeneous host configuration must be disabled for a given director/port/HBA and all flags must be reset.

To set (or reset) the HBA flags, use the following form:

```
symmask -sid SymmID -wwn wwn | -awwn awwn |
        -iscsi iscsi | -aiscsi aiscsi | -host HostName

set hba_flags
    <on <<flag>,<flag>...> <-enable | -disable> |
    off [<flag>,<flag>...] >
    -dir <#|all> -p <#|all>
```

Where:

**hba\_flags** — Sets the record in the database to hold information on the HBA port setting that may differ than the current setting on the corresponding FA.

**on | off** — Turns HBA flags on or off.

**flag** — Specifies the overridden HBA port flags from the values [in brackets]:

Common_Serial_Number [C]	AS400	[AS4]
Disable_Q_Reset_on_UA [D]	OpenVMS	[OVMS]
Environ_Set [E]	SCSI_3	[SC3]
Siemens [S]	Sunapee	[SCL]
Volume_Set_Addresssing [V]	Sequent	[SEQ]
Avoid_Reset_Broadcast [ARB]	SPC2_Protocol_Version [SPC2]	
	SCSI_Support1	[OS2007]



-enable — Enables the specified HBA port flag(s) on a per initiator basis.

-disable — Disables the specified HBA port flag(s) on a per initiator basis.

**Example** To turn on HBA flags and enable the `Common_Serial_Number` for host `api1182` on Symmetrix 343 for all directors and ports, enter:

```
symmask -sid 343 -host api1182 set hba_flags on C -enable -dir all -p all
```

Output similar to the following displays:

```
symmaskdb list db -sid 343 -v

Symmetrix ID           : 000190300343

Database Type          : Type6
Last updated at        : 08:18:30 AM on Tue May 19,2009
  Director Identification : FA-2D
Director Port          : 0

  Originator Port wwn    : 1111111111110002
  Type                   : Fibre
  User-generated Name    : api1182/001
  Visibility             : No
  FCID Lockdown          : No
  Lun Offset             : No
  Heterogeneous Host     : No
  Port Flag Overrides    : Yes
    Enabled              : Common_Serial_Number(C)
    Disabled             : N/A
  Dynamic Addressing     : No
  Authentication State   : N/A
  Devices                : 0098:009A

Director Identification : FA-2D
Director Port          : 1

  Originator Port wwn    : abcdefabcdef0001
  Type                   : Fibre
  User-generated Name    : api1182/002
  Visibility             : No
  FCID Lockdown          : No
  Lun Offset             : No
  Heterogeneous Host     : No
  Port Flag Overrides    : Yes
    Enabled              : Common_Serial_Number(C)
    Disabled             : N/A
  Dynamic Addressing     : No
  Authentication State   : N/A
  Devices                : 0044
```

## Setting the heterogeneous host configuration

---

**Note:** Setting the heterogeneous host configuration has been superseded by Setting the HBA port flags. The heterogeneous host configuration types listed in [Table 29](#) continue to be valid, but will not be expanded. To switch to setting HBA port flags, the heterogeneous host configuration must be disabled on the array, and all flags must be reset.

---

Heterogeneous host configuration is a feature that allows different host types to share a single director FA port even though they may require different port settings for their distinctive interface protocol.

Turning on heterogeneous host, turns on the bits listed in [Table 29](#), (and turns off the bits listed in footnote a) on the given director/port (for the given WWN or iSCSI). If this feature is enabled for one host type for a WWN, it must be disabled for that WWN before a new host type can be assigned.

This feature can be used in conjunction with the LUN offset skip feature to allow the different hosts their own LUN addressing scheme. With that scheme, the devices they see are different from those seen by any other host on the director.

The following syntax is used to set certain heterogeneous host configuration flags to optimize the host-to-director interface:

```
symmask -sid SymmID -wwn wwn -dir # -p # set heterogeneous on HostConfigFlag
```

Possible *HostConfigFlag* values are listed in the last column of [Table 29](#).

**Table 29** Host platforms and interface configuration flags (page 1 of 3)

Host platform	Requirements	Bit <sup>a</sup>	Host configuration flag ( <i>HostConfigFlag</i> )
AS/400		AS4	AS400
AS/400	Load source extender	AS4, V	AS400_LSE
Bull Escala/AIX			BULL_AIX
Bull Escala/AIX	PowerPath V1.5.x or earlier	D	BULL_AIX_PP15
Celerra		ARB	CELERRA
HP/DEC AlphaServers Tru64 UNIX 5.x FC-SW		OVMS	DEC_UNIX
HP/DEC OpenVMS		SC3, OVMS	DEC_OVMS
Data General AViiON NUMA 25000 Server		D	DG_AViiON
FSC BS2000/OSD Servers		C, D	FSC_2000
FSC PRIMEPOWER GP7000F Series host			PRIMEPOWER
FSC PRIMEPOWER GP7000F Series host	PowerPath V1.5.x or earlier	D	PRIMEPOWER_PP15
FSC PRIMEPOWER GP7000F Series host	VERITAS DMP	C, D	PRIMEPOWER_DMP
Fujitsu Services ICL Open VME		C	ICL_OPEN
Hewlett-Packard HP-UX		C, V	HP_UX
IBM AIX with FC 6227, 6228, 6239		SC3	IBM_AIX
IBM AIX with FC 6227, 6228, 6239	PowerPath V1.5.x or earlier	SC3, D	IBM_AIX_PP15
IBM AIX with FC 6227, 6228, 6239		C, SC3	IBM_AIX_DMP
IBM AIX with FC 6227, 6228, 6239		C, D, SC3	IBM_AIX_DMP_PP15
IBM AIX with EMC Fibre Channel			IBM_EMCC
IBM AIX with EMC Fibre Channel	PowerPath V1.5.x or earlier	D	IBM_EMCC_PP15
Linux			LINUX
Linux		C	LINUX_DMP

**Table 29** Host platforms and interface configuration flags (page 2 of 3)

Host platform	Requirements	Bit <sup>a</sup>	Host configuration flag ( <i>HostConfigFlag</i> )
Linux	Veritas Cluster Server	D, C, SC3	LINUX_DMP_VCS
NCR MP-RAS/Windows NT			NCR
NCR MP-RAS/Windows NT	Multiple vendor platforms	D	NCR_MP
NCR MP-RAS/Windows NT	If Windows NT is used with TNT, set FBA Env. Sense key to 4; otherwise, set it to 6	E	NCR_NT
NCR MP-RAS/Windows NT	Multiple vendor platforms: If Windows NT is used with TNT, set FBA Env. Sense key to 4; otherwise, set it to 6	D, E	NCR_NT_MP
Novell NetWare			NOVELL
Novell NetWare	Cluster	D	NOVELL_CLUSTER
Windows NT/Windows 2000			WINDOWS
Windows NT/Windows 2000	PowerPath V1.5.x or earlier	D	WINDOWS_PP15
Windows NT/Windows 2000	HP/Agilent controllers	V	WINDOWS_HP
Windows NT/Windows 2000	PowerPath V1.5.x or earlier, HP/Agilent controllers	D, V	WINDOWS_HP_PP15
Windows NT/Windows 2000	VERITAS VxVM DMP	C	WINDOWS_DMP
Windows NT/Windows 2000	HP/Agilent controllers VERITAS VxVM DMP	C, V	WINDOWS_HP_DMP
Windows NT/Windows 2000	PowerPath V1.5.x or earlier, VERITAS VxVM DMP	C, D	WINDOWS_DMP_PP15
Windows NT/Windows 2000	HP/Agilent and PowerPath V1.5.x, VERITAS VxVM DMP	C, D, V	WINDOWS_HP_DMP_PP15
Sequent NUMA-Q		E, C, SEQ	SEQUENT
Sequent NUMA-Q	FC-SW configurations only	E, C, SEQ, V	SEQUENT_FCSW
FSC Reliant UNIX RM series		E, D, S	RELIANT
Sun			SOLARIS
Sun	PowerPath V1.5.x or earlier	D	SOLARIS_PP15
Sun	VERITAS DMP	C, D	SOLARIS_DMP
Sun	Sun Cluster (ealier than 3.0)	C, D, SCL	SUN_CLUSTER
Sun	Sun Cluster (3.0 or higher)	C	SUN_CLUSTER30
VERITAS Cluster (VCS), EMC GeoSpan for VCS		D	VERITAS

**Table 29** Host platforms and interface configuration flags (page 3 of 3)

Host platform	Requirements	Bit <sup>a</sup>	Host configuration flag ( <i>HostConfigFlag</i> )
VERITAS Cluster (VCS), EMC GeoSpan for VCS	VERITAS DMP	C, D	VERITAS_DMP
VERITAS Cluster (VCS), EMC GeoSpan for VCS	VSC 2.0 or higher		VERITAS20
VMware (ESX 2.5 or earlier only)		C, SC3*	VMWARE

1. The following defines the host characteristic for each of the bits used in the table:

AS4	AS/400 secondary port
C	Common serial number for multipaths
D	Disable Queue Reset on Unit Attention (UA)
E	Environmental reports to host from Symmetrix
S	Enable Siemens host RM/400 - RM/600
SCL	Enable Sunapee (for Sun PDB clusters)
SC3	SCSI 3 interface
SEQ	Sequent Host (DYNIX/ptx)
OVMS	OpenVMS Fibre connection
V	Enable volume set addressing
ARB	Avoid Reset Broadcast
SPC2	SPC2 Protocol (Enginuity 5x71 and higher)
SC3*	Bit will only be set if Enginuity Version is lower than 5670
OS2007	Enable SCSI_Support1

# CHAPTER 12

## Device Masking: iSCSI Setup

Prior to using the device masking SYMCLI commands with iSCSI, you must configure your iSCSI driver software and authentication information. This chapter covers the following topics:

- ◆ [iSCSI authentication overview.....](#) 358
- ◆ [iSCSI CHAP authentication: Enginuity version 56xx .....](#) 358
- ◆ [iSCSI CHAP authentication: Enginuity version 57xx .....](#) 359
- ◆ [iSCSI software driver configuration.....](#) 363
- ◆ [RADIUS server configuration .....](#) 376

## iSCSI authentication overview

The iSCSI authentication is negotiated during the HBA login phase and can be implemented using CHAP. The implementation and configuration process for iSCSI CHAP authentication varies depending upon which Engenuity version is running on your host. Solutions Enabler supports the following Engenuity versions for iSCSI authentication:

- ◆ **Engenuity version 56xx:** For details on configuring iSCSI authentication, refer to [“iSCSI CHAP authentication: Engenuity version 56xx” on page 358](#).
- ◆ **Engenuity version 57xx:** For details on configuring iSCSI authentication, refer to [“iSCSI CHAP authentication: Engenuity version 57xx” on page 359](#).

In addition, Solutions Enabler provides support for a RADIUS (Remote Authentication Dial-In User Service) server for storing authentication information. For details on configuring your RADIUS server, refer to [“RADIUS server configuration” on page 376](#).

## iSCSI CHAP authentication: Engenuity version 56xx

CHAP (Challenge Handshake Authentication Protocol) allows you to manage a credential name and a CHAP secret, which are similar to a username and a password, though more secure than the standard Password Authentication Procedure (PAP).

### Requirements

Setting iSCSI authentication requires:

- ◆ Symmetrix DMX running Engenuity version 5670 (or higher)

---

**Note:** If you are running Engenuity version 57xx (5771 and higher), refer to [“iSCSI CHAP authentication: Engenuity version 57xx” on page 359](#).

---

- ◆ Fig-E board that manages the front-end connections in the Symmetrix array
- ◆ Gig-E-configured port
- ◆ Host systems that provide driver support for iSCSI
- ◆ An initialized device masking database (VCMDB)

Before an iSCSI host can log in and see any devices, the iSCSI name of the host must have a valid VCMDB database entry specifying the director and port from which it is connecting. For details on configuring your iSCSI drivers, refer to [“iSCSI software driver configuration” on page 363](#).

### Setting iSCSI authentication with the SYMCLI

Once you have configured your iSCSI driver software, and masked your devices, you can set the credential name and CHAP secret required for iSCSI authentication. The following command sets the authentication for the iSCSI initiator,

`iqn.2002-06.com.microsoft.host210`, using the authentication type of CHAP. The `-credential` option and `-secret` option specify the required authentication information:

```
symmask -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210
      set authentication -type CHAP -credential MyCredentials -secret
      MySecret
```

---

**Note:** To set iSCSI authentication for a Symmetrix array using Enginuity version 57xx (5771 and higher), requires the use of the `symconnect` command instead of the `symmask set authentication` command, which is valid only for Enginuity version 56xx (5670 and higher). For information about using `symconnect` commands, refer to [“iSCSI CHAP authentication: Enginuity version 57xx” on page 359](#).

---

## Showing authentication information

The `symmask show` command returns the authentication information for a specified iSCSI initiator on a specific Symmetrix array. Note that the CHAP secret is never displayed. For example:

```
symmask -sid 6208 -iscsi ign.2002-06.com.microsoft.host210  
show authentication
```

You can also display authentication data using the `symmaskdb list database -v` command.

When you set authentication using the `symmask` command, authentication is automatically enabled. If you decide to disable authentication, the authentication values are retained in the VCMDDB, which allows you to enable them at a later time using the `enable` action. The following command disables authentication:

```
symmask -sid 6208 -iscsi ign.2002-06.com.microsoft.host210  
disable authentication
```

## Restoring authentication data and updating the iSCSI driver

When you restore a database, authentication data is restored to the Symmetrix array from the backup file. Keep in mind that the same authentication data also needs to be set in the host's iSCSI driver software if authentication had been changed since the backup file was generated.

If you had changed your authentication data after the backup file was created, the restored authentication data will be out of sync with the authentication data stored in the iSCSI driver software. To rectify this discrepancy, you must update the iSCSI driver software with the same information contained in the restored authentication data. For more information, refer to [“iSCSI software driver configuration” on page 363](#).

To avoid restoring obsolete authentication data, use the `-skip_authentication` option. For example:

```
symmaskdb -sid 814 restore -file MyDevMaskBackup -skip_authentication
```

## iSCSI CHAP authentication: Enginuity version 57xx

Beginning with Solutions Enabler version 6.0.2 running on Symmetrix arrays using Enginuity version 5771, connection security for iSCSI ports is set with the `symconnect` command. The `symconnect` functionality allows you to set one-way or two-way CHAP authentication between a host HBA and a Symmetrix array.

If a RADIUS server is configured to store the authentication information, you can set the RADIUS server information and enable the Symmetrix array to look there for the authentication data. For more information on the RADIUS server, refer to [“RADIUS server configuration” on page 376](#).

## Setting one-way CHAP authentication

With CHAP one-way authentication, the array challenges the host during the initial link negotiation process and expects to receive a valid credential and CHAP secret in response. When challenged, the host transmits a CHAP credential and CHAP secret to the Symmetrix array. The Symmetrix array looks for this credential and CHAP secret in its own CHAP authentication database or on a RADIUS server (if one is set and turned on). Once a positive authentication occurs, the array sends an acceptance message to the host. However, if the array fails to find any record of the credential/secret pair, it sends a rejection message, and the link is closed.

Setting iSCSI authentication for a Symmetrix authenticator requires that you supply the Symmetrix ID, the iSCSI name, the CHAP credential name (username), and the CHAP secret (password). The following command inserts an entry for the iSCSI initiator (iqn.2002-06.com.microsoft.host210) into the Symmetrix CHAP authentication database. The `-cred` and `-secret` parameters specify the authentication data that the host should send in response to a challenge by the Symmetrix array.

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210
set chap -cred MyCredentials -secret MySecret
```

The CHAP protocol secret value (`MySecret` in this case) is a user-defined string up to 32 ASCII characters, or 64 binary characters (binary values should be prefixed with the string `0x`) for UNIX users. Windows users need to specify a secret between 12 and 16 characters and a credential name string between 8 and 256 characters.

The `set chap` action automatically enables CHAP authentication for the iSCSI initiator. You can use `disable chap` to disable CHAP authentication and `enable chap` to turn it back on.

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210
disable chap
```

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210
enable chap
```

Use the `delete chap` command to disable authentication, as follows:

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210
delete chap
```

## Setting two-way CHAP authentication

With two-way CHAP authentication, the host challenges and authenticates the Symmetrix too. Thus, both the host and Symmetrix array act as authenticators, and both act as peers. Configuring two-way authentication requires that you configure a one-way authentication for each communication direction.

[“Setting one-way CHAP authentication” on page 360](#) shows how to set up the Symmetrix array as an authenticator, which covers one direction of communication. This section describes how to configure authentication from the opposite direction.

The following command illustrates how to set up the Symmetrix array as a peer. In this case we must define what credential and secret the Symmetrix array will send when challenged by the host HBA authenticator:

```
symconnect -sid 6208 -dir 2D -p 0 set chap -cred MyPeerCredentials
-secret MyPeerSecret
```



The `-dir` and `-p` parameters specify the director (2D) and port (0) on the Symmetrix array (6208) through which a host HBA is connected. Make a note of the `-secret` value set here. This value will have to be set in the iSCSI Initiator properties window as described in [“Configuring iSCSI for two-way chap authentication” on page 370](#).

---

**Note:** The host authenticator must have a user authentication database that contains matching authentication data for the authentication to be successful.

---

The `set chap` command automatically enables CHAP authentication for the Symmetrix director and port. You can use `disable chap` to disable CHAP authentication and `enable chap` to turn it back on:

```
symconnect -sid 6208 -dir 2D -p 0 disable chap
symconnect -sid 6208 -dir 2D -p 0 enable chap
```

Use the `disable chap` command to disable authentication, as follows:

```
symconnect -sid 6208 -dir 2D -p 0 delete chap
```

## Initializing an authentication database

In most cases, you do not want to clear the data of an existing authentication database. If you are unsure whether an authentication database currently exists for a Symmetrix array, issue a command that displays any existing Symmetrix CHAP or RADIUS authentication database for the array. For example, to view any authentication database if it exists on Symmetrix 6208:

```
symconnect -sid 6208 -list
```

To initialize and clear an authentication database, you must specify a backup file name to safeguard against clearing data in the database that should not be lost. For example, the following commands create backup files for Symmetrix CHAP and RADIUS authentication databases and attempt to write any current authentication data to these backup files prior to initializing:

```
symconnect -sid 6208 init chap -file MyChapInitBackup
symconnect -sid 6208 init radius -file MyRadiusInitBackup
```

## Displaying authentication information

You can use the basic `symconnect list` command to display a Symmetrix array's authentication database (for example, any Symmetrix CHAP or RADIUS authentication database for Symmetrix 20):

```
symconnect -sid 20 list
```

To display just the Symmetrix CHAP authentication database, include the `chap` option. For example:

```
symconnect -sid 20 list chap

Symmetrix ID           : 000190300020

Director Identification : SE-2D
Director Port          : 0

Protocol               : CHAP

Identifier              Type      State      Credential
-----
```

```

SE-2D:0                                N/A      ENABLED  symm20
iqn.2002-06.com.microsoft.host210 iSCSI  DISABLED api210
iqn.2002-06.com.microsoft.host211 iSCSI  ENABLED  api211

```

CHAP credentials have been defined in the Symmetrix authentication database for two iSCSI initiators, making it possible for the Symmetrix to authenticate either iSCSI HBA logging into its port. Credentials have also been defined for the Symmetrix director and port (SE-2D:0) in the event that two-way authentication is required and the Symmetrix array needs to submit authentication data to the host.

To display only a RADIUS authentication database, include the `radius` parameter. For example:

```
symconnect -sid 20 list radius
```

```

Symmetrix ID           : 000190300020

Director Identification : SE-2D
Director Port          : 0

Protocol               : RADIUS_SERVER

Server Rank  State      Server Name      IP Address      Port
-----
Primary      ENABLED    AP22             108.15.139.220  1812 <- default
1st Backup   ENABLED    AP23             108.15.139.221  1812 <- default
2nd Backup   ENABLED    AP24             108.15.139.222  1812 <- default

Protocol           : RADIUS
Identifier          Type      State
-----
iqn.2002-06.com.microsoft.host210 iSCSI  DISABLED
iqn.2002-06.com.microsoft.host211 iSCSI  ENABLED

```

## Backing up and restoring authentication information

Backing up authentication information on a regular basis ensures that you can restore a good version of the authentication database in case incorrect changes or other abnormalities occur in the current database.

The following command creates a backup version of the Symmetrix user authentication database in a file called `MyChapBackup` (each time you back up the database, you must specify a new file name):

```
symconnect -sid 6208 -file MyChapBackup backup chap
```

The following command restores this authentication database using the backup version in the file `MyChapBackup`:

```
symconnect -sid 6208 -file MyChapBackup restore chap
```

Similarly, you can back up and restore the authentication database on a RADIUS server by specifying `backup radius` and `restore radius`.

To display the contents of a backup authentication file, use the `symconnect list` command with the filename:

```
symconnect list -file MyChapBackup
```

# iSCSI software driver configuration

This section provides the requirements and configuration steps for preparing a host system with a native iSCSI initiator and the VCMDB (residing in a Symmetrix DMX) to communicate over the Microsoft iSCSI software driver and the EMC Multi-Protocol Channel Director.

The following is an overview of the configuration process:

1. Collect information about the host computer and the Symmetrix DMX Multi-Protocol Channel director:
  - a. Get the iSCSI name and IP address of the Symmetrix director from the Symmetrix array.
  - b. Get the iSCSI name of the initiator from the host computer.
2. Configure information in the VCMDB of the Symmetrix array that will allow the host computer to access the Symmetrix devices desired:
  - a. Add access to the Symmetrix devices from the host initiator by the iSCSI name to the VCMDB.
  - b. Add the iSCSI authentication information (if any) about the host initiator.
  - c. Refresh the database.
3. Update the iSCSI initiator with the Symmetrix information:
  - a. Establish a target connection using iSCSI between the host computer and the designated Symmetrix director port with/without authentication.
  - b. Have the host computer logon to the Symmetrix over iSCSI and establish target devices on the host computer that will persist through a reboot.
4. Perform the following disk administration on the devices on the Symmetrix array, if needed:
  - a. Format
  - b. Write signatures
  - c. Assign drive letters

Configuration of the iSCSI driver can be completed with CHAP authentication as described in “[Configuring iSCSI with CHAP authentication](#)” on page 364, or without authentication as described in “[Configuring iSCSI without CHAP authentication](#)” on page 371. Refer to the procedure that meets your environment’s authentication requirements.

## Requirements

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

---

**Note:** Your systems must be installed and connected to the network before beginning the configuration.

---

## Installing the iSCSI software initiator

From the Microsoft download site, download and install Microsoft iSCSI Software Initiator. The setup installs an icon on your desktop.

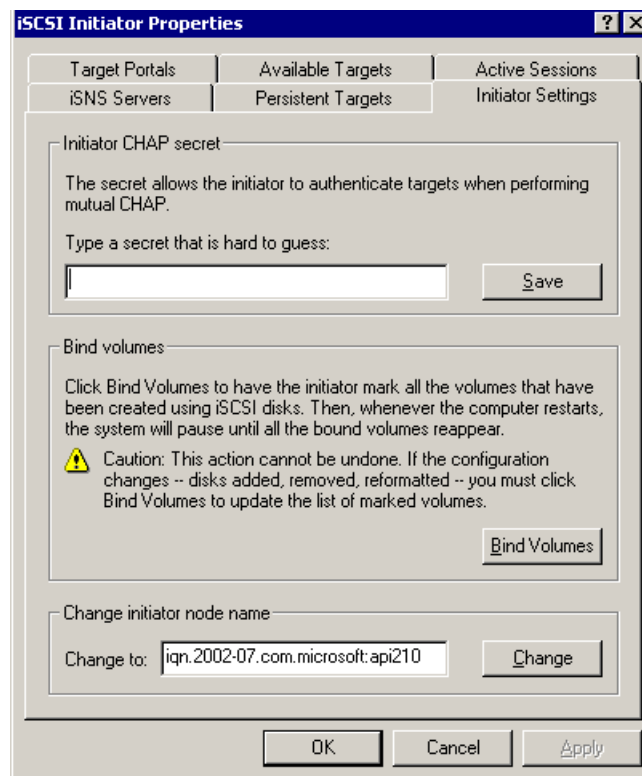
## Configuring iSCSI with CHAP authentication

To complete this configuration, you must obtain the iSCSI name of the host, and the iSCSI ID of the multi-protocol director.

**Note:** This procedure contains steps that require the use of the iSCSI Initiator window and the DOS command window, as follows.

### Collect information about the host computer and Symmetrix director

1. Execute the iSCSI Initiator control panel on the host system.
2. Click the **Initiator Settings** tab.



3. Copy the iSCSI ID of your host from the **Change to** field.

**Note:** Do not enter any data in this window.

4. From the command line, display and copy the IP address of the Symmetrix multi-protocol director, as shown in this example:

```
symcfg -sid 6208 -dir 3a list -v
```

The last two lines of the display contain the iSCSI name and IP address. For example:

```
Symmetrix ID: 000000006208
Product Model                : DMX2000P
```

```

Symmetrix ID                : 000000006208
Microcode Version (Number)  : 5670 (16260000)
.
iSCSI NAME                   : iqn.1992-04.com.emc.5006048000061002
iSCSI IP Address             : 10.10.10.21

```

### Configure information in the VCMDB

5. Add a device to create a record in the VCMDB, using the following form:

```
symmask -sid SymmID -iscsi iscsi -dir # -p # add dev #
```

Where:

*SymmID* — Identifies the Symmetrix ID.  
*iscsi* — Specifies the iSCSI name (from step 3).  
 -dir # — Identifies the Symmetrix director number.  
 -p # — Identifies the Symmetrix port number.  
 add dev # — Specifies the Symmetrix device number(s).

*Example:* **symmask -sid 6208 -iscsi iqn.2002-07.com.microsoft:api210 -dir 3a -p 0 add dev 0023**

6. Set the CHAP authentication in the VCMDB using the SYMCLI, which is done differently for each supported Enginuity version:

- For Enginuity version **5771** or higher:

```
symconnect -sid SymmID -iscsi iscsi
set chap -cred Credential -secret Secret
```

Where:

*SymmID* — Identifies the Symmetrix ID.  
*iscsi* — Specifies the host iSCSI name.  
*Credential* — Uses between 8 and 256 alphanumeric characters.  
*Secret* — UNIX: 32 ASCII characters, or 64 binary characters (binary values should be prefixed with the string 0x).  
 Windows: 12 and 16 ASCII characters.

- For Enginuity version **5670** or higher:

```
symmask -sid SymmID -iscsi iscsi
set authentication -type CHAP
-credential Credential -secret Secret
```

Where:

*SymmID* — Identifies the Symmetrix ID.  
*iscsi* — Specifies the host iSCSI name.  
 CHAP — Indicates the authentication type.  
 CHAPcredential — Uses 8 alphanumeric characters.  
 CHAPsecret — Uses 12-16 alphanumeric characters.

---

**Note:** For details about setting iSCSI Authentication, refer to [“iSCSI CHAP authentication: Enginuity version 56xx” on page 358](#) or [“iSCSI CHAP authentication: Enginuity version 57xx” on page 359](#).

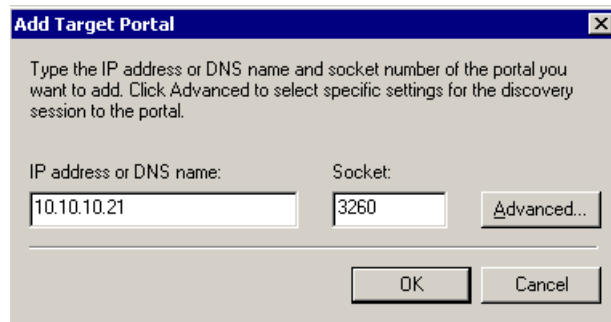
---

7. Refresh the VCMDB, as shown in the following example:

```
symmask -sid 6208 refresh
```

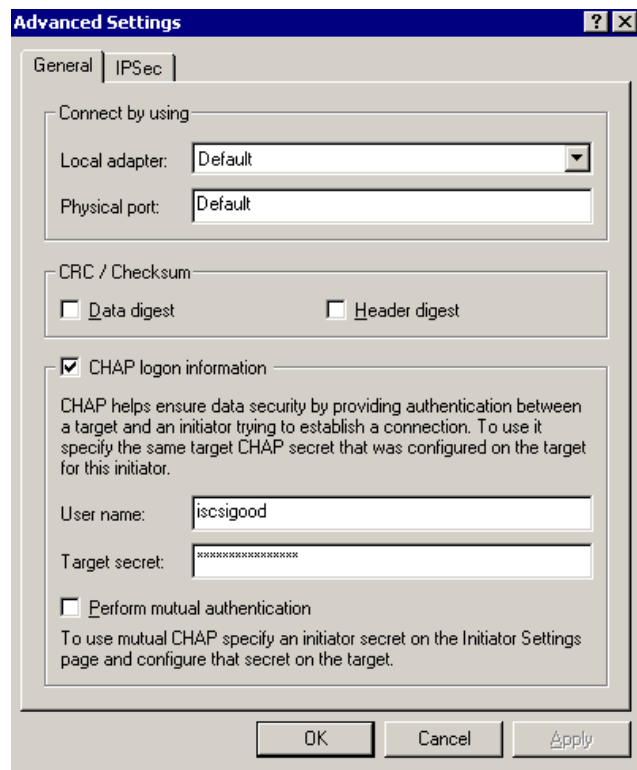
### Update iSCSI initiator with Symmetrix information

8. From the iSCSI Initiator Properties window, click the **Target Portals** tab and click **Add**. The **Add Target Portal** dialog box appears.



9. Enter the iSCSI IP address (from step 4) of the Symmetrix multi-protocol director in the **IP address or DNS name** box and click **Advanced**.

The **Advanced Settings** dialog box appears. The iSCSI name automatically displays in the **User name** field. Do not use this for your credential.



10. Check the box labeled **CHAP logon information**. Change the CHAP credential in the **User name** field, and enter a CHAP secret in the **Target secret** box. Click **OK**.

The credential name string must be between 8 and 256 characters. The CHAP protocol secret value on UNIX can be 32 ASCII characters, or 64 binary characters (binary values should be prefixed with the string 0x). On Windows the secret must be between 12 and 16 ASCII characters.

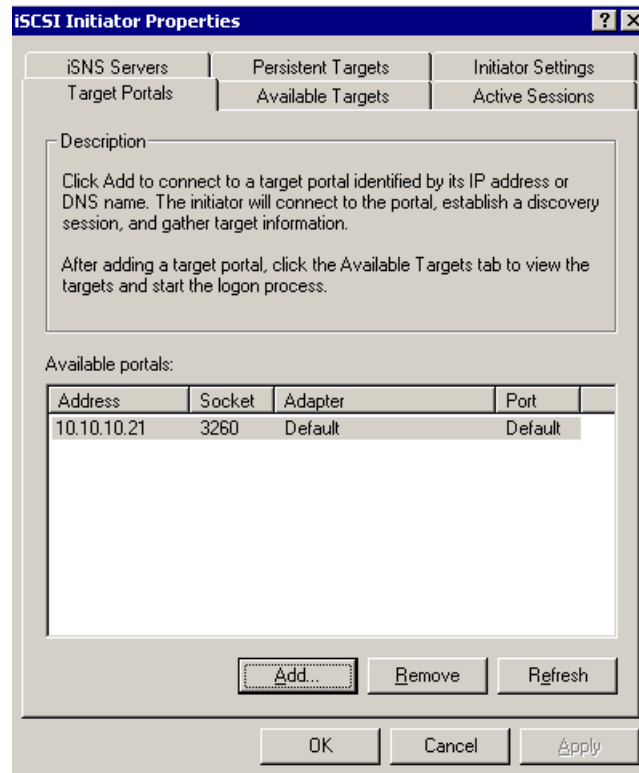
---

**Note:** Do not check or change anything else in this dialog box.

---

11. Click **OK** to close the **Advanced Settings** dialog box, and click **OK** to close the **Target Portal** dialog box.

The multi-protocol director IP address should appear in the **Available portals** list in the **Target Portals** window.

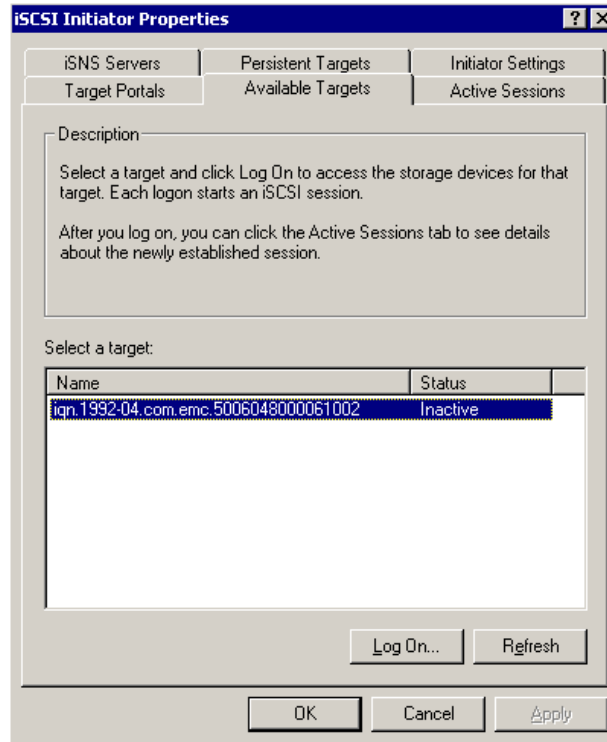



---

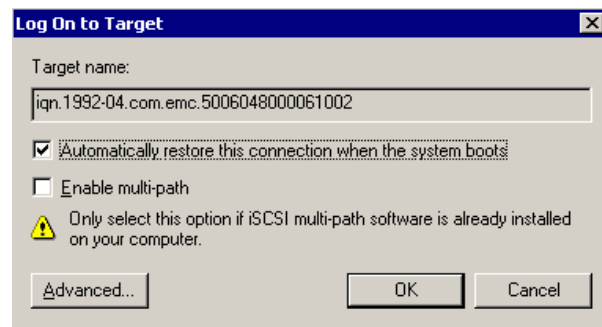
**Note:** If an error displays, select the IP address from the **Available portals** list and click **Remove**. Begin the configuration procedure again.

---

12. Click the **Available Targets** tab. The iSCSI name of the Symmetrix multi-protocol director (from step 4) displays in the **Select a target** list.



13. Click **Log On**. The **Log On to Target** window dialog box appears.

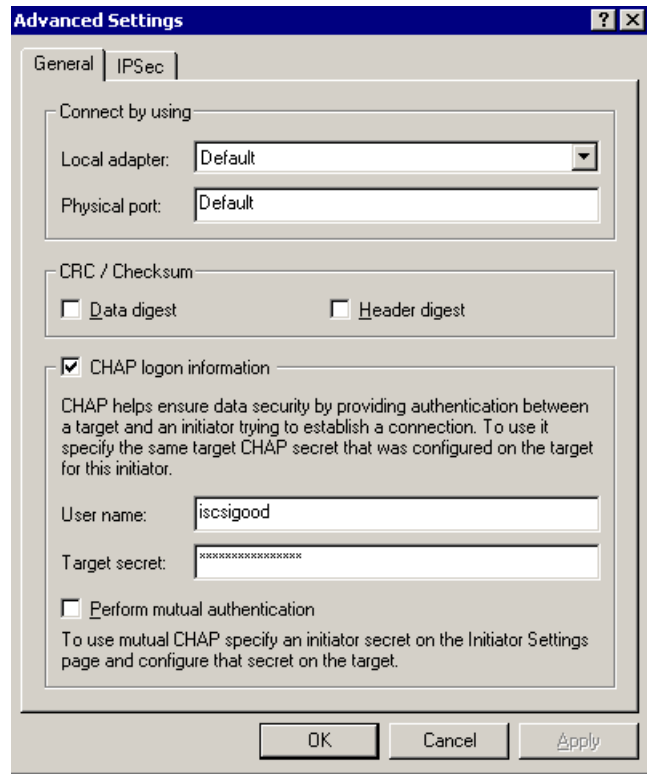


14. Check **Automatically restore this connection when the system boots**, and click **Advanced**.

The **Advanced Settings** dialog box appears. The iSCSI name automatically displays in the **User name** field.

Do not use this for your credential.





15. Select the box labeled **CHAP logon information**. Change the CHAP credential in the **User name** field, and enter a CHAP secret in the **Target secret** box. Click **OK**.

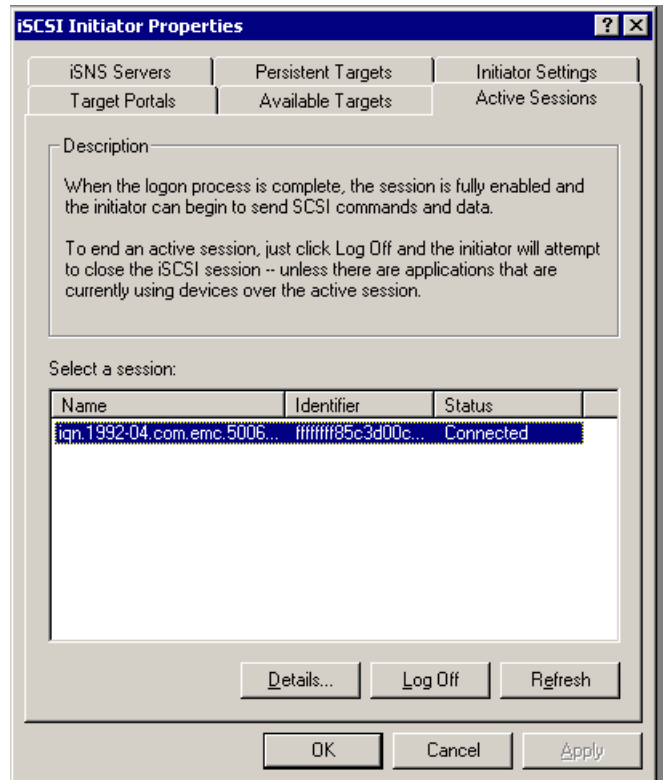
The credential name string must be between 8 and 256 characters. The CHAP protocol secret value on UNIX can be 32 ASCII characters, or 64 binary characters (binary values should be prefixed with the string 0x). On Windows the secret value must be between 12 and 16 ASCII characters.

---

**Note:** Do not select or change anything else in this window.

---

16. From the iSCSI Initiator Properties window, click the **Active Sessions** tab. An active session should display in the **Select a session** list.



17. Select the session and click **Details** to display the disks found by the iSCSI driver.
18. Click **OK** to exit from the iSCSI Initiator.

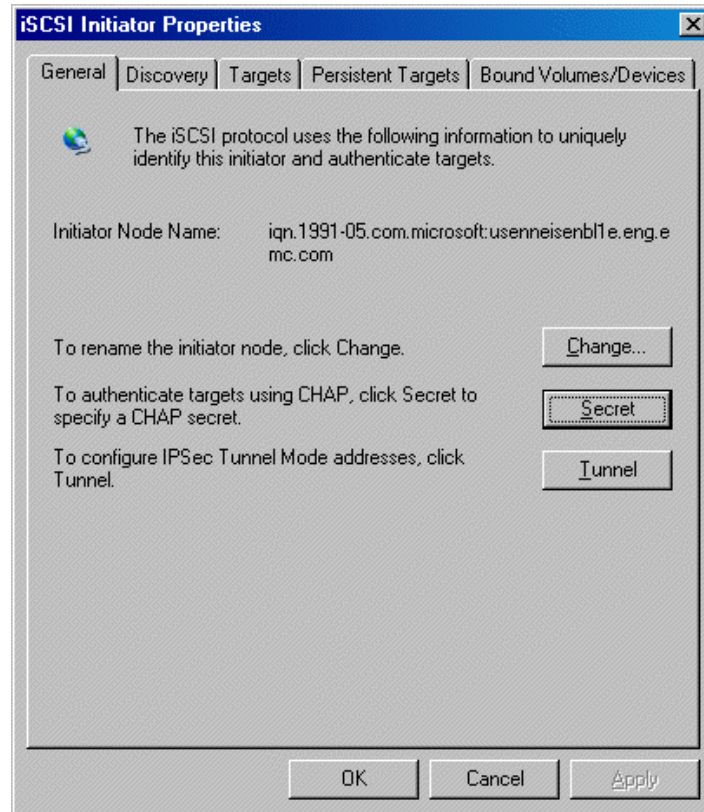
#### Perform disk administration on the Symmetrix devices

19. Perform any necessary disk administration, such as, formatting, write signatures, and assigning drive letters.
20. Reboot the host system.

**Note:** The configuration changes you made will not take effect until you reboot your host.

## Configuring iSCSI for two-way chap authentication

For a two-way Chap configuration, after installing your iSCSI Initiator, open your iSCSI Initiator properties, click the **Secret** button on the **General** tab, and set the secret for two-way CHAP authentication to match the value set using `symconnect` as described in [“Setting two-way CHAP authentication” on page 360](#).



## Configuring iSCSI without CHAP authentication

To complete this configuration, you must obtain the iSCSI name of the host, and the iSCSI ID of the multi-protocol director.

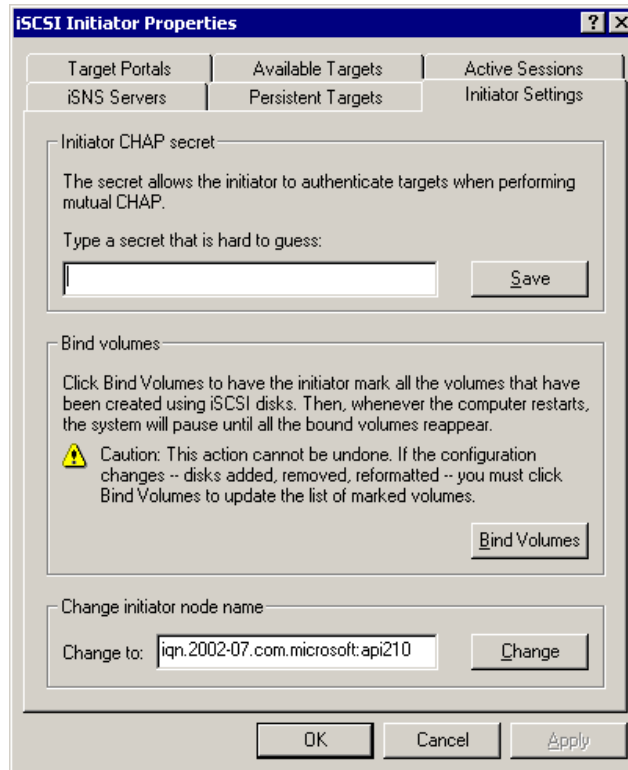
---

**Note:** This procedure contains steps that require the use of the iSCSI Initiator Properties window and the SYMCLI command window, as follows.

---

### Collect information about the host computer and Symmetrix director

1. Open the iSCSI Initiator on the host system.
2. Click the **Initiator Settings** tab.



- Copy the iSCSI ID of your host from the **Change to** field at the bottom of the window.

**Note:** Do not enter any data in this window.

- List the iSCSI name and IP address of the Symmetrix multi-protocol director, as shown in this example:

```
symmcfg -sid 6208 -dir 3a list -v
```

The last two lines of the display contain the iSCSI name and IP address. For example:

```
Symmetrix ID: 000000006208
Product Model           : DMX2000P
Symmetrix ID            : 000000006208
Microcode Version (Number) : 5670 (16260000)
.
iSCSI NAME              : iqn.1992-04.com.emc.5006048000061002
iSCSI IP Address        : 10.10.10.21
```

#### Configure information in the VCMDB

- From the command line, add a device to create a record in the VCMDB, using the following form:

```
symmask -sid SymmID -iscsi iscsi -dir # -p # add dev #
```

Where:

*SymmID* — The Symmetrix ID.

*iscsi* — The iSCSI name copied in step 3.

*-dir #* — Symmetrix director number.

*-p #* — Symmetrix port number.

`add dev #` — Symmetrix device number.

For example:

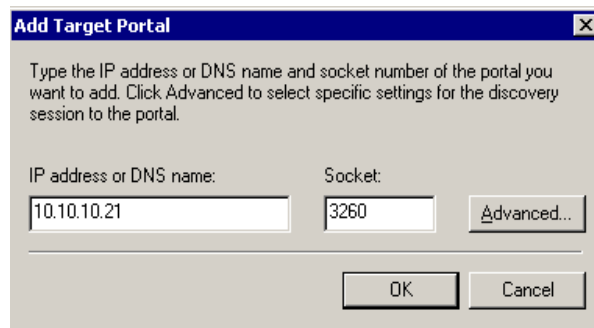
```
symmask -sid 6208 -iscsi ign.2002-07.com.microsoft:api210 -dir 3a -p 0  
add dev 0023
```

6. Refresh the VCMDB, as shown in the following example:

```
symmask -sid 6208 refresh
```

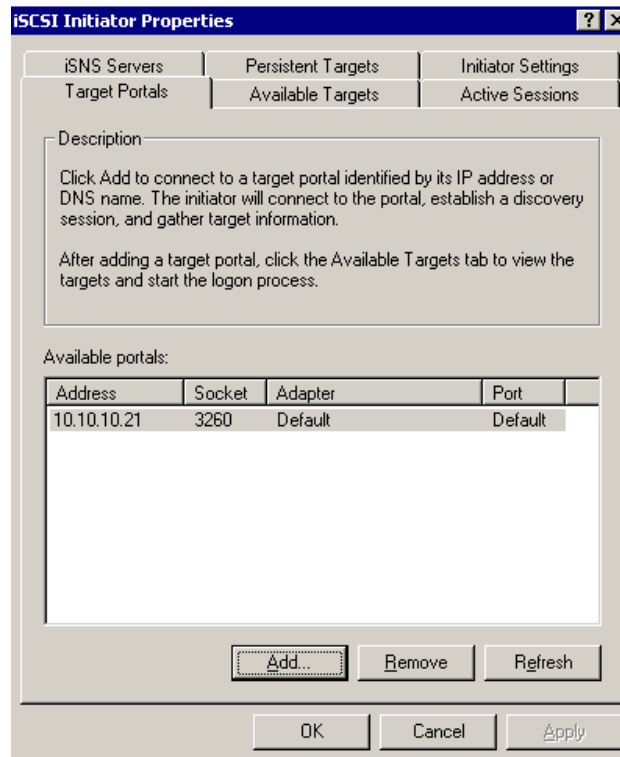
7. From the iSCSI Initiator window, select the **Target Portals** tab and click **Add**.

The **Add Target Portal** dialog box appears.



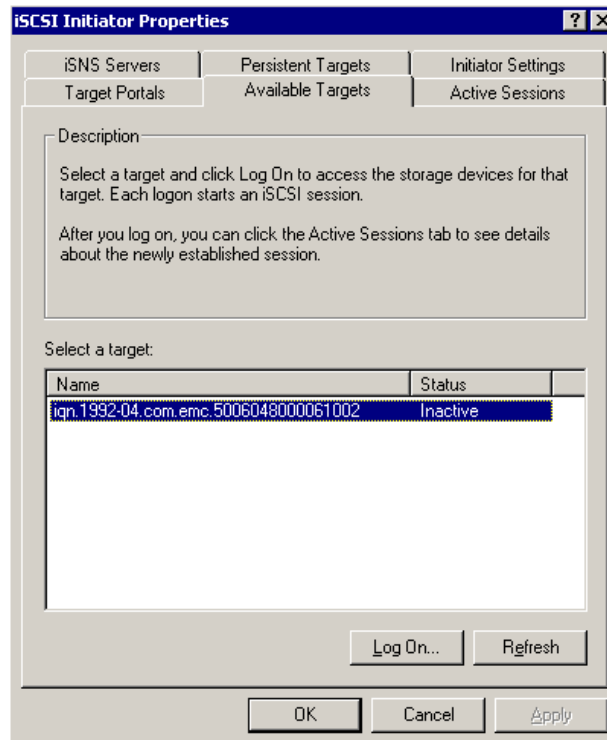
8. Enter the iSCSI IP address (from step 4) of the Symmetrix multi-protocol director in the **IP address or DNS name** box and click **OK**.

The multi-protocol director IP address should appear in the **Available portals** list in the **Target Portals** window.

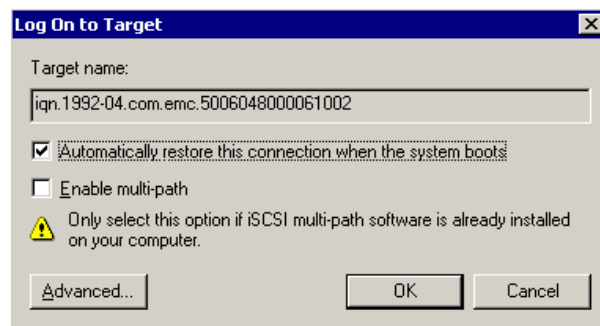


**Note:** If an error displays, select the IP address from the **Available portals** list and click **Remove**. Begin the configuration procedure again.

9. Click the **Available Targets** tab. The iSCSI name of the Symmetrix multi-protocol director (from step 4) appears in the **Select a target** list.

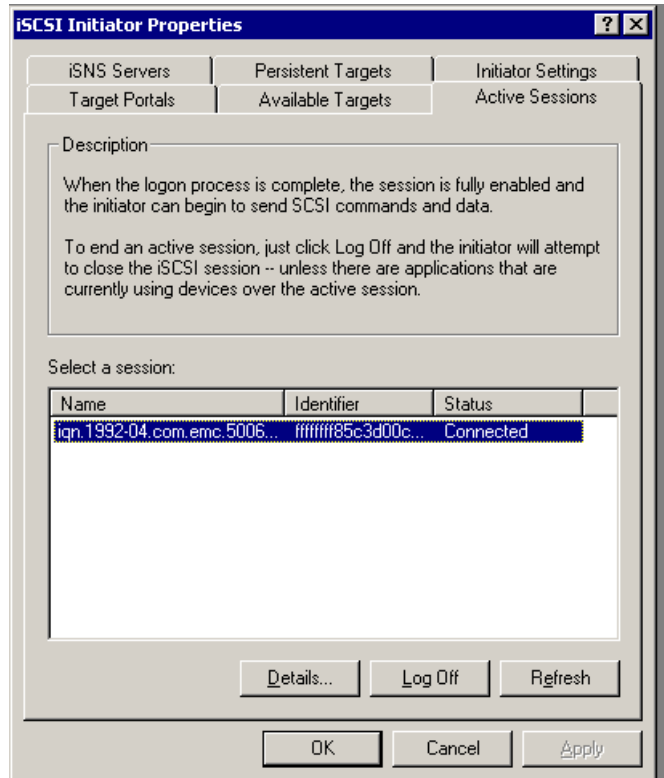


10. Click **Log On**. The **Log On to Target** dialog box appears.



11. Select **Automatically restore this connection when the system boots**, and click **OK**.

12. Click the **Active Sessions** tab. An active session should display in the **Select a session** list.



13. Select the session and click **Details** to display the disks found by the iSCSI driver.

14. Click **OK** to exit from the **iSCSI Initiator Properties** window.

#### Perform disk administration on the Symmetrix devices

15. Perform any necessary disk administration, such as, formatting, write signatures, and assigning drive letters.

16. Reboot the host system.

**Note:** The configuration changes you made will not take effect until you reboot your host.

## RADIUS server configuration

When using a RADIUS server to store authentication data, you need to use the `symconnect` command to make the SYMAPI server aware of your RADIUS server. Once registered, the Symmetrix SYMAPI server will look to the RADIUS server for matching authentication data when attempting to authenticate a host's credential and secret.

You set the RADIUS server information at the Symmetrix director/port level, letting the Symmetrix know where to go for the authentication information when a host HBA tries to log in. The required information includes the server name, its IP address, its key (a password of up to 256 characters), and its rank (`primary`, `backup1`, or `backup2`). For example:

```
symconnect -sid 6208 -dir 2D -p 0 set radius -server AP22
-key MyServerPassword -ip 108.15.139.220 -rank primary
```



The `-rank` parameter establishes server priority preference if two or three RADIUS servers are specified for a given director-port combination. If a primary and two backup servers are enabled, the system tries them in the obvious order until successful (a server might be unreachable for some reason). If primary is disabled, the system looks to `backup1` and `backup2`.

If the RADIUS server port to be used is different than the server's default port, you can specify it using the `-port` option. You can also include options that specify a server retry interval in seconds (`-i`) and the number of times to retry (`-c`).

The `set radius` command automatically enables RADIUS authentication for the Symmetrix array's director and port. You can use `disable radius` to disable authentication and `enable radius` to turn it back on:

```
symconnect -sid 6208 -dir 2D -p 0 disable radius -rank primary  
symconnect -sid 6208 -dir 2D -p 0 enable radius -rank primary
```

Once you have set and enabled all RADIUS server information and at the Symmetrix director and port level, you can then enable the server authentication feature for a given iSCSI HBA logging into a port. Use either `set radius` or `enable radius` to turn on the RADIUS server authentication for the specified iSCSI initiator. For example:

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210  
enable radius
```

Similarly, you can use `disable radius` to turn off RADIUS authentication for the iSCSI initiator.

Use `delete radius` to delete the RADIUS authentication entry, as follows:

```
symconnect -sid 6208 -iscsi iqn.2002-06.com.microsoft.host210  
delete radius
```



# CHAPTER 13

## Managing Network IPsec

This chapter describes the IPsec standard and how to manage IPsec network policies, using the IPsec component of the SYMCLI. The chapter covers the following topics:

◆ Introduction .....	380
◆ IPsec and Symmetrix.....	381
◆ SYMCLI IPsec management component .....	382
◆ Viewing policy information .....	383
◆ Command file scripting for creating/modifying policies .....	387
◆ Committing a policy-scripted command file .....	392
◆ Deleting a policy-scripted command file .....	392

# Introduction

This section overviews the IPsec standard and how it applies to Symmetrix networks.

## IPsec standard

IPsec is a framework of open standards for security of network communication that provides data confidentiality, integrity, and authentication between participating peers at the IP layer of an Ethernet network.

IPsec allows two peers (computers) to trust each other's identity and share a private key for each communication session. Based on the session key, the computer peers establish an encrypted communications channel and can verify that data they receive is what was originally sent. These countermeasures mitigate many security attacks, including spoofing, tampering with data en route, connection hijacking, eavesdropping, and replay of transmission. IPsec is the same security technology used in many corporate VPN solutions.

IPsec technology protects the Internet Protocol (IP) version 4 and 6 (IPv4 and IPv6) standards. IPv4 is the most commonly used, but IPv6, sometimes called IPng (next generation) can provide more addresses, better security, reduced administrative workload, and enhanced support for mobile devices.

More specifically, the IPsec framework is an Internet Protocol standard (RFC 2401) that allows you to create IP network tunnels through an existing IP network. All traffic contained within this tunnel is afforded some configurable measure of protection against hostile exterior entities. This is accomplished by optionally transforming each network packet at the tunnel source in some way, wrapping the result with an IPsec header, and sending this wrapped packet to the destination. The process is reversed at the destination. First, the IPsec wrapper is stripped off, and the packet is transformed again. It then appears as it originally did, and may be processed by conventional means, as if it had never been modified.

IPsec offers three main services to users:

- ◆ Authentication — Verifies that a packet was originated by the sender
- ◆ Integrity — Ensures that packets are not modified in transit
- ◆ Encryption — Hides the packet contents from prying eyes

Not all of these services are necessary or even desirable at the same time, in all situations. Users may selectively employ the ones that fit their current needs.

## IPsec policies

An IPsec implementation internally manages database records called policies. They are processed in order, sorting on an attached index number called the policy's priority. In general, policies describe how to select applicable network traffic, and how to apply IPsec protection to it. They contain proposal and transform requirements for the IP tunnel.

More specifically, a policy is a combination of parameters that describe how network traffic will be encrypted. A policy is a single record in the policy database, and is indexed/sorted by Priority. When network traffic appears that might be affected by encryption, the IPsec implementation scans forward, in priority order, through the policy database looking for a policy that matches the traffic. Each policy has proposals and

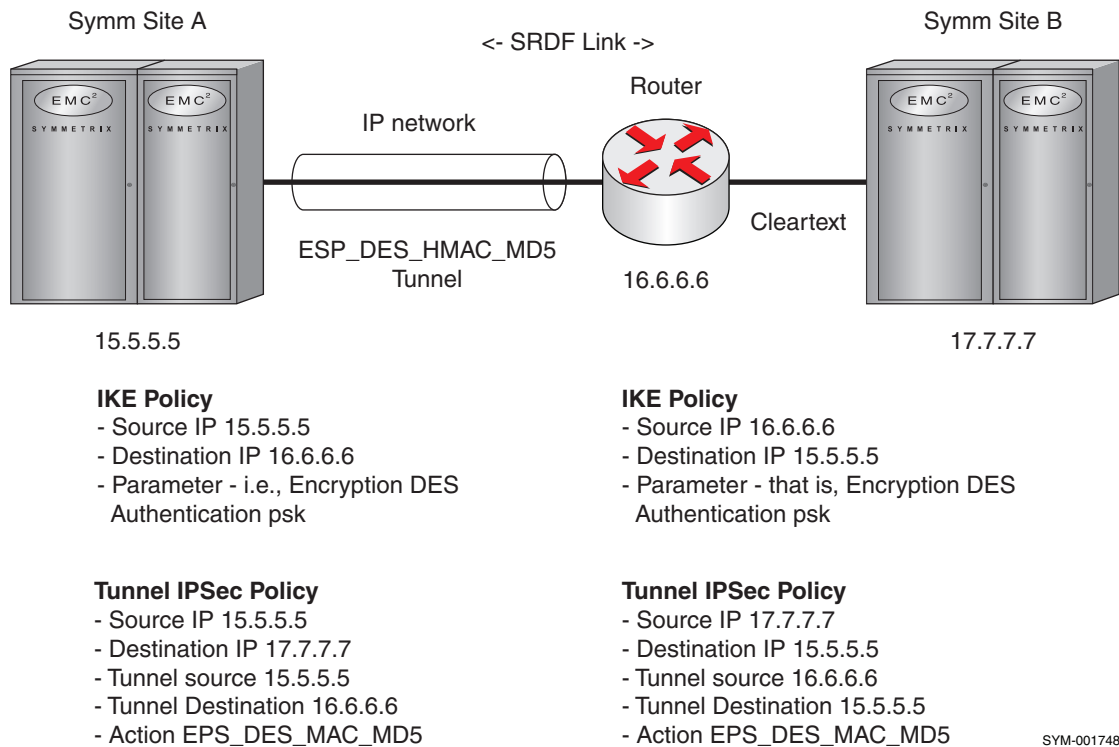
transforms that are also tested, in order. Once a match has been found, searching is terminated, and an entry is created in the Security Associations database that reflects the policy found. Also, each policy has a unique priority.

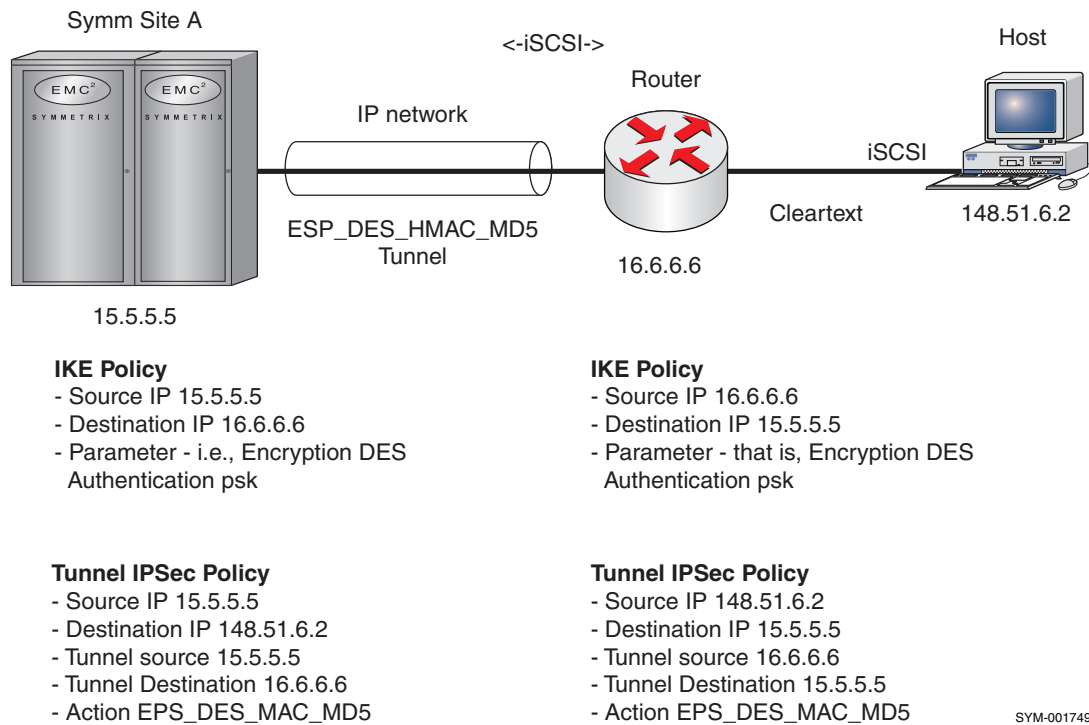
## IPsec and Symmetrix

The Symmetrix subsystem, running Enginuity 5773 and higher, in companion with specific crypto processors, provides IPsec traffic services on each of its Gigabit Ethernet (Advanced Multi-Protocol channel) directors. Here, IP network traffic to or from the array can be protected by IPsec applied to it that supports two significant network connection protocols:

- ◆ SRDF — Symmetrix to Symmetrix endpoints, securing SRDF links
- ◆ iSCSI — iSCSI host to Symmetrix endpoints or NAS to Symmetrix endpoints

Figure 35 illustrates an SRDF IPsec link and Figure 36 shows a host endpoint with the iSCSI connection.



**Figure 35** IPsec network tunnel with Symmetrix-to-Symmetrix endpoints**Figure 36** IPsec network tunnel with iSCSI host-to-Symmetrix endpoints

Traffic selection within IPsec is extremely fine-grained, as it is possible to apply IPsec only to iSCSI traffic with a connection to a particular host, or series of hosts, leaving other iSCSI traffic and SRDF untouched. Different destinations can have different types of protection.

**Note:** With such vast flexibility and versatility, configuration of IPsec is considered somewhat complex. It is recommended that IPsec management, described in this chapter, and the *EMC Solutions Enabler Symmetrix CLI Command Reference* be performed with or by network engineers that understand encryption disciplines and the standard RFC 2401 for IPsec, including details of your specific network and environment. Incorrect intervention in this area, could expose critical information to third parties; especially if the two endpoints are connected with a publicly-visible network.

For more information about the RFC 2401 standard, see the following:

<http://rfc.net/rfc2401.html>

## SYMCLI IPsec management component

The SYMCLI IPsec component extends the basic SYMCLI command set to include the Solutions Enabler `symipsec` command. This command allows you to perform all the necessary control operations that manage various types of IPsec security association policy changes in your Symmetrix environment. This IPsec standards compliant command provides policy management to/from the IPsec processor chip on the Symmetrix Gig-E port. It also offers retrieval of policies and associated statistics.

If you employ the Access Control feature with `symacl`, access types BASE and CFGSYM are required.

Information retrieved from the IPsec subsystem is not stored in the SYMAPI database. The IPsec policies are stored in global memory and backed up to the Symmetrix vault, therefore policy database backups and restores are not provided.

The `symipsec` command allows you to display and set values of control parameters for Symmetrix IPsec encryption and authentication support as follows:

- ◆ Lists various active IPsec policies.
- ◆ Shows detail about an active policy.
- ◆ Lists the various statistics and errors in the Enginuity processing of policies.
- ◆ Previews and commits the scripted command file that defines the specific network security policy requirements.
- ◆ Uses command file entries to define the IPsec policy requirements, including details such as proposals, and transforms.

## Viewing policy information

With the `symipsec` command, you can return a list of IPsec policies or show detail about a policy being processed on a director and port.

---

**Note:** Currently, when specifying port numbers in the `symipsec` command, only port 0 is a valid parameter (and is the default). Specifically targeting Port 1 on a director is not supported.

---

### Listing policies

To list information about policies on a specified Symmetrix array and director, use the following form:

```
symipsec -sid SymmID list -dir #|ALL -port 0
        -priority Level#
        -all
```

Where:

*Level#* — the policy level index number. The lower the number, the higher the priority level (0-110).

For example, to list the policy priority numbers on director 16D within a specific Symmetrix array, enter:

```
symipsec -sid 343 -dir 16D list port 0 -all
```

The output will list all current policies on the specified director:

```
Symmetrix ID           : 000190300343
Director number        : 16D
Port number            : 0
Policy priorities present :
    0          10      20          30          80          90
    100        110
```

## Showing policy detail

To show detail information about policies on a specified Symmetrix array, use the following form:

```
symipsec -sid SymmID show -dir #|ALL -port #|ALL
          -priority Level#
          -all
```

For example, to show the policy details on director 1A for a specific policy (20), enter:

```
symipsec -sid 123456789012 -dir 1A -port 0 show -priority 20
```

For a second example, to show the policy details on director 16D about all current policies, enter:

```
symipsec -sid 343 -dir 16D show -all
```

The output will show detail information about all current policies on the specified director, for example:

```
Symmetrix ID           : 000190300343
Director ID           : 16D
Port number           : 0
Policy priority       : 0
Action                : Secure
Local Endpoint:
  Endpoint type        : Single
  IP Version           : v4
  IP Address           : 10.0.0.40
  IP Port Number       : ANY
  IP Protocol Number   : ANY
  Flags                : N/A
Remote Endpoint:
  Endpoint type        : Single
  IP Version           : v4
  IP Address           : 10.0.0.51
  IP Port Number       : ANY
  IP Protocol Number   : ANY
  Flags                : N/A
Source Port Selectivity : Packet
Protocol Selectivity    : Policy
Destination Port Selectivity : Policy
Proposal Set:
  Proposal Set Type     : IKE
  IN SPI                : 0
  OUT SPI               : 0
  IN CPI                : 0
  OUT CPI               : 0
  IN Nonce              : 0
  OUT Nonce             : 0
  IKE Mode              : Main
  PFS                   : Off
Number of Proposals     : 1
  Proposal #            : 1
  Proposal type :       : IKE
Number of Transforms    : 1
  Transform #           : 1
  Transform type        : IKE
  Authorization Algorithm : MD5
  Encryption Algorithm   : 3DES
  Lifetime (bytes)       : 0
  Lifetime (seconds)     : 86400
  ESP Mode               : Transport
  DH Group               : DH Group 1
  Authentication Method  : Preshared Key
```



```

        Compression Algorithm      : N/A
Symmetrix ID                      : 000190300343
Director ID                      : 16D
Port number                      : 0
Policy priority                  : 10
Action                          : Secure
Local Endpoint:
.
.
.

```

The output pattern repeats for every policy found on the specified director.

## Listing policy statistics

To list statistical information about the IPsec processing of a policy or policies, use the following form. You can optionally target a remote endpoint address:

```

symipsec -sid SymmID list -dir #|ALL -port 0
          -stats -type StatsType [-priority Level#][-remote_addr IPendPt]

```

Where:

*StatsType* — possible values are:

- ike\_errors
- ipsec\_details

*IPendPt* — the IP endpoint address

*Level#* — the policy level index number. The lower the number, the higher the priority level (0-110).

## Examples for retrieving statistical details

To display IPsec SA statistical details for a specific priority (20) on director 1A, enter:

```

symipsec -sid 123456789012 -dir 1A -port 0 list -stats -type
ipsec_details -priority 20

```

For a second example, to display IPsec SA statistical details for a specified director 2C, enter:

```

symipsec -sid 067 -dir 2c list -stats -type ipsec_details

```

The output will list statistical detail information about all current policies on the specified director, for example:

```

Statistics from director 02C, port 0
      Local endpoint
Endpoint type      : Single
IP Version         : v6
IP Address         : fd00:abcd:0:0:0:0:1:50
IP Port Number     : 1748
IP Protocol Number : ANY
Flags              : N/A
      Remote endpoint
Endpoint type      : Single
IP Version         : v6
IP Address         : fd00:abcd:0:0:0:0:1:41
IP Port Number     : 48235
IP Protocol Number : ANY
Flags              : N/A

SA Type            : ESP
ESP Mode           : Transport

```

```

Encryption Algorithm      : AES 128
Authentication Algorithm  : XCBC
Compression Algorithm     : N/A
Total Rekeys              : 0

```

#### Inbound SA values

```

SPI                      : 4025390203
Byte Limit                : 1099511627776
Seconds Limit            : 5400
Packet Count             : 0
Compressed Packet Count  : 0
User Byte Count          : 0
Compressed User Byte Count : 0
Authentication Errors    : 0
Replay Errors            : 0
Padding Errors           : 0

```

#### Outbound SA values

```

SPI                      : 4066949324
Byte Limit                : 1099511627776
Seconds Limit            : 5400
Packet Count             : 0
Compressed Packet Count  : 0
User Byte Count          : 0
Compressed User Byte Count : 0

```

Statistics from director 02C, port 0

Local endpoint

```
Endpoint type           : Single
```

```

.
.
.

```

The output pattern repeats for every policy found on the specified director.

## Examples for retrieving IPsec errors on the director security processor

To display IPsec IKE errors in the security processor on director 2C, enter:

```
symipsec -sid 067 -dir 2c list -stats -type ike_errors
```

The output will list statistical IKE or IPsec errors related to any of the current policies on the specified director, for example:

Statistics from director 34, port 0

```

Endpoint type           : Single
IP Version              : v6
IP Address              : fd00:abcd:0:0:0:0:1:50
IP Port Number          : ANY
IP Protocol Number      : ANY
Flags                   : N/A

```

#### Phase 1 (IKE) errors

```

Init Failures :          : 0
Response Failures :       : 0
Invalid Cookies :         : 0
No Peer Responses :       : 0
Invalid Peer Responses :   : 0
Logical Failures :        : 0

```

#### Phase 2 (IPsec) errors

```

Init Failures :          : 0
Response Failures :       : 0
Invalid Cookies :         : 0
No Peer Responses :       : 0
Invalid Peer Responses :   : 0
Logical Failures :        : 0

```

## Command file scripting for creating/modifying policies

The command and control of IPsec policy declarations are done through scripting and managing a command file. You can add, modify, or delete policies using various entries in a specified command file.

### Command file syntax adding/modifying policies

The following is the command file syntax to add a network policy or modify an existing network policy. Leading pound signs (#) on a command line entry comments out that notation from the command parser:

---

**Note:** You can only define one proposal and one transform per policy declaration.

---

```
policy add|modify
  -priority Level#
  -action secure|discard|bypass
  [-assoc_ike_policy Level#]
  #only if proposal_type is ipsec
  -local_addr IPAddr
    [-ipproto IPprotocol#|all][-ipport IPport#|all][-mask IPAddr]
  -remote_addr IPAddr
    [-ipproto IPprotocol#|all][-ipport IPport#|all][-mask IPAddr]
  [-remote_tunnel_addr IPAddr]
  #only if esp_mode is tunnel
  [-selectivity {destip|destport|srcport} SPECIFICITY]
  #for policies with wildcarded ipaddr/port/proto only

  -proposal_set
  -proposal_set_type auto|ike>manual
  [-ike_mode main|aggressive]
  #if proposal_set_type is set to ike mode
  [-pfs on|off]
  #if proposal_set_type is set to ike mode
  [-key_format hex |ascii]
  [-presharedkey Keystring]
  #if proposal_set_type is set to ike mode
  [-inenc_key Keystring -outenc_key Keystring]
  #if transform type is ESP or IKE and proposal is manual mode
  [-inauth_key Keystring -outauth_key Keystring]
  #if proposal is set to manual mode and algorithm is not null
  [-in_spi SPI# -out_spi SPI#]
  #if proposal is set to manual mode and algorithm is AES mode
  [-in_nonce NONCE -out_nonce NONCE]
  #if proposal is set to manual mode

  -proposal
  -proposal_type ike|ipsec

  -transform
  -transform_type ike|esp
  [-auth_alg null|sha1|md5|xcbc]
  [-enc_alg null|des|3des|aes_128|aes_256|aes_cm_128|aes_cm_256]
  #if transform type is esp or ike
  [-dhgroup 1|2|3|4]
  #if transform type is ike
  [-esp_mode tunnel|transport]
  #if transform type is esp
  [-lifetime [LifeParam1][,][LifeParam2]]
  [-auth_method preshared_key|dsa|rsa]
  #if transform type is ike. DSA and RSA not supported.

;
```

## Scripting the policy syntax

The beginning section of the policy syntax sets the overall policy requirements. For a policy add or modify task, consider and set the following options:

```
-priority Level#
-action secure|discard|bypass
[-assoc_ike_policy Level#]
-local_addr IPAddr
  [-ipproto IPprotocol#|all] [-ipport IPport#|all] [-mask IPAddr]
-remote_addr IPAddr
  [-ipproto IPprotocol#|all] [-ipport IPport#|all] [-mask IPAddr]
[-remote_tunnel_addr IPAddr]
[-selectivity {destip|destport|srcport} SPECIFICITY]
```

Where:

`-priority Level#`

Specifies the index number of the policy to be added or modified. Possible priorities range from 0 to 110. When packets arrive, policies with lower numbered priorities are examined first. Also, any associated IKE policies (`-assoc_ike_policy`) must have a lower priority index number (higher priority) than this object IPsec policy.

`-action secure|discard|bypass`

Sets the policy action to `secure`, requests that IPsec processing will be fully applied to packets that match this policy, requiring packet transformation.

Sets the policy action to `discard`, requests that the matching packets will be dropped completely. For use on unauthorized break-ins.

Sets the policy action to `bypass`, requests that the matching packets will be completely unmodified, and pass straight through the IPsec processor without encryption.

`-assoc_ike_policy Level#`

Specifies the index number of the associated IKE policy to the object IPsec policy. The `-proposal_type` option must be set to `ipsec`. When packets arrive, policies with lower numbered priorities are examined first. Associated IKE policies must have a lower priority index number (higher priority) than the corresponding IPsec policy.

`-local_addr IPAddr`

```
  [-ipproto IPprotocol#|all] [-ipport IPport#|all] [-mask IPAddr]
-remote_addr IPAddr
  [-ipproto IPprotocol#|all] [-ipport IPport#|all] [-mask IPAddr]
```

These options primarily define the local and remote endpoint IP addresses.

Optionally, you can target an IP protocol number (`IPprotocol#`) or all IP protocols and/or an IP port number (`IPport#`) or all IP port numbers. For example, IP protocol 6 should be used for TCP and 1 for IPv4-ICMP. And for example, IP port 1748 should be used for SRDF and port 3260 for iSCSI host connections. Note the SYMCLI can detect and handle both IPv4 and IPv6 address notations (dots and/or colons) in the command entry.

`-remote_tunnel_addr IPAddr`

Specifies the IP address of the remote tunnel object. The `-esp_mode` option must be set to `tunnel`.

`-selectivity {destip|destport|srcport} SPECIFICITY`

You can choose to use selectivity lists confined to destination or source points and can include *SPECIFICITY* types/protocols for wildcarded proposals only. When an endpoint field's properties have been wildcarded, it determines whether new connections will share an existing security association (selectivity *POLICY*), or if new connections will cause a new security association to be created (selectivity *PACKET*). Selecting *PACKET* results in a more secure configuration, since encryption keys won't be shared between connections, but consumes more resources. Selecting *POLICY* conserves security associations, when this is desired. Properties that may be wildcarded include IP address, IP port number, and IP protocol number.

## Scripting proposal set syntax

The proposal set section of the policy syntax sets the proposal settings requirements. For a policy add or modify, consider and set the following options:

```
-proposal_set
-proposal_set_type auto|manual|ike
[-ike_mode main|aggressive]
[-pfs on|off]
[-key_format hex |ascii]
[-presharedkey Keystring]
[-inenc_key Keystring -outenc_key Keystring]
[-inauth_key Keystring -outauth_key Keystring]
[-in_spi SPI# -out_spi SPI#]
[-in_cpi CPI# -out_CPI CPI#]
[-in_nonce NONCE -out_nonce NONCE]
```

Where:

```
-proposal_set
-proposal_set_type auto|manual|ike
```

These options start the proposal set declaration. The `proposal_set_type` can be set to `auto`, `manual`, or `ike` for key management.

Automatic key management or SA management protocol is required to support anti-replay features of AH and ESP modes, and to accommodate on-demand creation of SAs. Multiple keys can be created as a result of this operation.

Manual key management is the simplest form of management where you must manually configure each system with keying data and security association management data relevant to secure communication with other systems. Manual techniques are practical in small, static environments, but it does not scale well.

The Internet Key Exchange (IKE) protocol is a key management protocol standard which is used in conjunction with the IPsec standard. IKE is a hybrid protocol which implements the Oakley key exchange and Skeme key exchange inside the SA and key management protocol (ISAKMP) framework. IKE eliminates the need to manually specify all the IPsec parameters in the crypto maps at both peer endpoints. It allows lifetime limits, changes to occur in keys during IPsec sessions, and allows anti-replay services and dynamic authentication of peers.

## IKE options

```
-ike_mode main|aggressive
-pfs on|off
-presharedkey Keystring
```

These options require `proposal_set_type` to be set to `ike`.

IKE phase 1 negotiations are used to establish IKE security associations (SAs). These SAs protect the IKE 2 negotiations. IKE uses one of two modes for phase 1 negotiations: `main` or `aggressive`. Main mode is more intense and secure, but time consuming. Aggressive mode provides faster negotiations, but exposes identities of the peers to eavesdropping.

Perfect Forward Secrecy (PFS) mode for IKE policies can be turned on or off. (Typically, this should be left on, unless you have a special environment.

The preshared key option (`-presharedkey`) specifies the preshared key as the authentication method and defines the preshared key string. It should be the same secret string shared between security points.

## Manual options

```
-inenc_key Keystring -outenc_key Keystring
-inauth_key Keystring -outauth_key Keystring
-in_spi SPI# -out_spi SPI#
-in_cpi CPI# -out_CPI CPI#
-in_nonce NONCE -out_nonce NONCE
```

These options require `proposal_set_type` to be set to `manual`.

For non-null `-enc_alg` only, the `-inenc_key` and `-outenc_key` options specify an encryption key string used for encrypting/decrypting inbound and outbound traffic. These must match the corresponding field on the remote endpoint.

For non-null `-auth_alg` only, the `-inauth_key` and `-outauth_key` options specify a hash key string used for authenticating inbound and outbound traffic. These must match the corresponding field on the remote endpoint.

The `-in_spi` and `-out_spi` options specify a Security Parameter Indexing (SPI) number for inbound and outbound traffic in security associations decode.

The `-in_nonce` and `-out_nonce` options specify a random nonce value for inbound and outbound traffic to counter replay attacks.

## Proposal options

```
-proposal
-proposal_type ike|ipsec
```

These options start the proposal declaration. Currently, only one proposal per policy can be declared. Compression mode is not supported.

The `proposal_type` specifies the type of proposal to apply to the policy. Possible types supported are:

```
ike — IP Key Exchange
ipsec — IPsec
```

## Transform options

```
-transform
-transform_type ike|ipsec
```

These options start the proposal declaration. Currently, only one transform per proposal and policy can be declared.

The `transform_type` specifies the type of transform to apply to the policy proposal. Possible types supported are:

- `ike` — IP Key Exchange
- `esp` — Encapsulation Security Payload

Currently, `ah` for authentication header is not supported.

```
-auth_alg null|sha1|md5|xcbc
```

Specifies the authentication algorithm for IKE or ESP policy and transform hash functions. Possible values are:

- `null`
- `sha1` — SHA-1 hash authentication algorithm
- `md5` — MD5 hash authentication algorithm
- `xcbc` — AES-XCBC-MAC constant key authentication algorithm

```
-enc_alg null|des|3des|aes_128|aes_256|aes_cm_128|aes_cm_256
```

Specifies the encryption algorithm for IKE or ESP policy transforms. Possible values are:

- `null`
- `des` — DES encryption algorithm
- `3des` — 3DES or Triple DES (a.k.a. TDES) encryption algorithm
- `aes_128` — Advanced Encryption Standard (AES) encryption (128-bit)
- `aes_256` — Advanced Encryption Standard (AES) encryption (256-bit)
- `aes_cm_128` — Advanced Encryption Standard counter mode (AES-CM)
- `aes_cm_256` — Advanced Encryption Standard counter mode (AES-CM)

```
-dhgroup 1|2|3|4
```

Specifies which Diffie-Hellman (`dh`) group to use for the symmetrical key generation for IKE transforms only. Groups 1 through 4 are supported.

```
-esp_mode tunnel|transport
```

Specifies the ESP transform mode. Possible values are:

- `tunnel` — Entire IP packet is encrypted/authenticated (full secure)
- `transport` — Only the payload (just data) of the IP packet is authenticated

```
-lifetime [LifeParam1][,][LifeParam2]
```

Specifies the life of the security policy with time and/or data size parameters for IKE and ESP only. Just one, or both values in any order, can be specified:

- A time value in minutes or hours (such as, 90 min or 5 hr)
- A data size value in megabytes or gigabytes (such as, 50 MB or 3 GB)

If both are specified, apply a comma between parameters with not intervening space. The first limit reached, will end the policy life.

```
-auth_method preshared_key
```

Specifies the authentication method for IKE transforms. Possible values are:

```
preshared_key
```

DSA and RSA methods are not supported.

## Committing a policy-scripted command file

Once you have completed scripting the policy in the command file. You need to commit the file for execution. However, due to extensive notation work in any policy command file, it is recommended to preview the file for command syntax errors.

### Preview the file

To preview the IPsec policy syntax of a command file, use the following command form:

```
symipsec -sid SymmID -dir # -port 0 -file /tmp/commandfile preview
```

**Note:** When specifying port numbers in the `symipsec` command, only port 0 is a valid parameter (and is the default). Specifically targeting Port 1 on a director is not supported.

The following example, targets director 1A to preview an IPsec policy addition to the policy database, using file `/tmp/ap`:

```
symipsec -sid 0039 -dir 1A -port 0 -file /tmp/ap preview
```

### Commit the file

To commit the IPsec policy syntax of a command file, use the following command form:

```
symipsec -sid SymmID -dir # -port 0 -file /tmp/commandfile commit
```

The following example, targets director 1A to commit an IPsec iSCSI policy addition to the policy database, using file `/tmp/ap`:

```
symipsec -sid 0039 -dir 1A -port 0 -file /tmp/ap commit
```

Where `/tmp/ap` contains:

```
policy add -priority 50 -assoc_ike_policy 40 -action secure
-local_addr 172.23.195.20 -ipport 3260 -ipproto 6
-remote_addr 50.60.70.80 -ipport 3260 -ipproto 6
-selectivity destip packet -selectivity destport packet
-proposal_set -proposal_set_type auto -proposal
-proposal_type ipsec -transform -transform_type esp
-enc_alg aes_cm_256 -lifetime 90m,5gb;
```

## Deleting a policy-scripted command file

You can delete an existing IPsec policy.

To remove an IPsec policy from the policy database, enter:

```
symipsec -sid 0039 -dir 1A -port 0 -file /tmp/dp commit
```

Where `/tmp/dp` contains:

```
policy delete -priority 50;
```



# PART 2

## Operational Examples

The Operational Examples part of this product guide identifies and focuses on some specific array control tasks that represent the most typical practices in the management of your Symmetrix storage environment. These practical examples illustrate various array control processes by showing the SYMCLI command sequences to accomplish these tasks. These specific management tasks are described in the subsequent chapters as follows:

[Appendix A, “Configuration Change Examples,”](#) provides detailed examples using the Configuration Change component.

[Appendix B, “Auto-provisioning Groups Example,”](#) provides an example of creating an initiator group, port group, storage group, and masking view.

[Appendix C, “FAST Output Examples,”](#) provides several examples of FAST output and explains the column headings.

[Appendix D, “Virtual LUN Migration Example,”](#) provides a detailed example of a virtual LUN migration.

[Appendix E, “Updating the Host,”](#) explains how to update a host so that the host recognizes the new Symmetrix configuration.

[Appendix F, “SIU: Overview and Management,”](#) describes the Symmetrix Integration Utilities (SIU) disk management operations for your storage system in a Windows Server environment.

[Appendix G, “Managing Legacy Time Windows,”](#) explains how to manage time windows using the `symoptmz` command.



# APPENDIX A

## Configuration Change Examples

This appendix contains examples using the configuration change action. These examples illustrate advanced configuration concepts. Only advanced users or system administrators should change a Symmetrix configuration.

- ◆ [Creating devices.....](#) 396
- ◆ [Mapping devices.....](#) 398
- ◆ [Setting front-end port attributes.....](#) 402
- ◆ [Configuring concurrent SRDF devices.....](#) 406
- ◆ [Updating a disk label with emulated device geometry .....](#) 412
- ◆ [Configuring Virtual Provisioning .....](#) 420
- ◆ [Automatically creating metadevices .....](#) 427
- ◆ [Setting port flags on a front-end director .....](#) 430

## Creating devices

This section provides an example of creating a device on a Symmetrix VMAX Family array running Enginuity V5874.

For detailed information about a Symmetrix configuration, use `symconfigure list` with the verbose (`-v`) option, as shown in the following sample command and output:

**symconfigure -sid 1135 list -v**

```

Symmetrix ID                      : 000192601135

Configuration Server Version      : 5874.240.191
Configuration Server Protocol     : 0xA09
Configuration Server Date        : 09.01.2010

FBA formatted                     : 2009727
FBA unformatted                   : 0
AS/400 formatted                 : 1273392
AS/400 unformatted               : 0
IBM_FBA formatted                : 2009727
IBM_FBA unformatted              : 0
UNISYS_FBA formatted            : 2009727
UNISYS_FBA unformatted           : 0
ICL formatted                    : 0
ICL unformatted                  : 0
CKD-3380 formatted              : 0
CKD-3380 unformatted             : 0
CKD-3390 formatted              : 2294762
CKD-3390 unformatted             : 0

Max Hypers per Disk              : N/A
FBA Multi Access Cache           : N/A
Dynamic RDF Configuration        : Enabled
RAID-S support                   : N/A
RAID membership count            : N/A
VCMdb Access Restricted           : N/A
RAID-5 support                   : Enabled

```

The `syndisk list` command displays the capacity of the disk, as shown in the following sample command and output:

**syndisk -sid 1135 list |less**

```

Symmetrix ID                      : 000192601135
Disks Selected                    : 250

```

Ident	Symb	Int	TID	Vendor	Type	Hypr	Total	Capacity(MB)	
								Free	Actual
DF-7A	07A	C	0	SEAGATE	T146155	339	139028	14293	139028
DF-7A	07A	C	2	SEAGATE	T146155	340	139028	7261	139028
DF-7A	07A	C	4	SEAGATE	T146155	340	139028	6643	139028
DF-7A	07A	C	6	SEAGATE	T146155	0	0	0	139028
.									
.									
DF-10D	10D	D	13	SEAGATE	T146155	113	139028	17080	139028
DF-10D	10D	D	15	SEAGATE	T146155	115	139028	24286	139028
							-----	-----	-----
Totals							32504697	3126809	35982751

The following command creates one 2-Way-Mir FBA device in disk group 3:

```
symconfigure -sid 397 -cmd "create dev count=1, size=220, config=2-Way-Mir,  
emulation=FBA, disk_group=3;" commit
```

Execute a symconfigure operation for symmetrix '000194900397' (y/[n]) ? y

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000194900397
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 004 of 046 steps.....Executing.
Step 014 of 046 steps.....Executing.
Step 023 of 046 steps.....Executing.
Step 025 of 156 steps.....Executing.
Step 042 of 156 steps.....Executing.
Step 042 of 156 steps.....Executing.
Step 076 of 156 steps.....Executing.
Step 077 of 156 steps.....Executing.
Step 080 of 156 steps.....Executing.
Step 082 of 156 steps.....Executing.
Step 085 of 156 steps.....Executing.
Step 085 of 156 steps.....Executing.
Step 094 of 156 steps.....Executing.
Step 094 of 156 steps.....Executing.
Step 111 of 156 steps.....Executing.
Step 117 of 156 steps.....Executing.
Step 118 of 156 steps.....Executing.
Step 135 of 156 steps.....Executing.
Step 135 of 156 steps.....Executing.
Step 140 of 156 steps.....Executing.
Step 141 of 156 steps.....Executing.
Step 156 of 156 steps.....Executing.
Local: COMMIT.....Done.
```

New symdev: 159C

Terminating the configuration change session.....Done.

To monitor the configuration change session while the `symconfigure commit` operation is processing, issue the `symconfigure query` command, as shown in [“Setting front-end port attributes” on page 402](#).

When you add new devices, the SYMAPI host database file on the host issuing the configuration change is updated automatically on successful completion of the `symconfig commit` command. (Automatic update of the local SYMAPI database occurs for all configuration changes except mapping changes.) To refresh the SYMAPI host database files on all *other* hosts attached to that Symmetrix array, you can perform the `symcfg sync` command on those hosts.

To confirm that the new devices have been added correctly, issue another `symdev list` command.

## Mapping devices

The hardware setup remains the same as the previous example: an HP-UX host (api157) and two Solaris hosts (api145 and api146), each connected to a Symmetrix array (sid 120). All commands are issued from Solaris host api145. The example maps the four new Symmetrix devices to front-end director FA-4B so that these devices can be made visible to host api145.

Using the `-connections` option with `symcfg list` allows you to see the host connections to a Symmetrix array connected to a host running SYMAPI software. The following display shows the front-end directors that each host uses to reach the Symmetrix (sid 120). Although a host can use more than one front-end director to connect to the same Symmetrix array, host api145 here uses only director FA-4B. The “FA” prefix indicates a fibre front-end adapter, while an “SA” prefix indicates a SCSI front-end adapter:

```
symcfg list -connections -sid 120
```

```
Symmetrix ID : 000185500120
```

Symmetrix		Host				
Director	Port	Node Name	IP Address	HW Type	OS Name	OS Revision
FA-5B	0	api146	172.23.65.146	sun4u	SunOS	5.7
FA-4B	0	api145	172.23.65.145	sun4u	SunOS	5.6
SA-14A	1	api157	172.23.65.157	9000/897	HPUX	B.11.00

The `symdev list` command with the `-noport` option displays devices that are not yet mapped to any front-end (SA) adapter ports, including the newly-created devices (009B through 009E):

```
symdev list -sid 120 -noport
```

```
Symmetrix ID: 000184500120
```

Device Name		Directors		Device			
Sym	Physical	SA :P	DA :IT	Config	Attribute	Sts	Cap (MB)
0000	Not Visible	?????	01A:C0	Unprotected	N/Grp'd	RW	103
000A	Not Visible	?????	02B:C0	BCV	N/Asst'd	RW	103
000B	Not Visible	?????	15B:C4	Unprotected	N/Grp'd (M)	RW	309
000E	Not Visible	?????	16B:C0	Unprotected	N/Grp'd	RW	103
000F	Not Visible	?????	01B:C4	Unprotected	N/Grp'd	RW	103
0010	Not Visible	?????	01A:D0	Unprotected	N/Grp'd (M)	RW	516
0015	Not Visible	?????	02A:C4	2-Way Mir	N/Grp'd	RW	103
0016	Not Visible	?????	16A:D0	3-Way Mir	N/Grp'd	RW	103
0017	Not Visible	?????	01A:C4	4-Way Mir	N/Grp'd	RW	103
0099	Not Visible	?????	15B:C1	Unprotected	N/Grp'd	RW	103
009A	Not Visible	?????	01A:C3	Unprotected	N/Grp'd	RW	103
009B	Not Visible	?????	01A:C5	2-Way Mir	N/Grp'd	RW	103
009C	Not Visible	?????	02B:C1	2-Way Mir	N/Grp'd	RW	103
009D	Not Visible	?????	16A:C4	2-Way Mir	N/Grp'd	RW	103
009E	Not Visible	?????	15B:C3	2-Way Mir	N/Grp'd	RW	103

The `sympd list` command displays devices that are visible to this host, meaning that these devices are currently the only ones on the Symmetrix array that have been mapped to a front-end director (04B) and recognized by performing a host update:

```
sympd list -sid 120
```

Symmetrix ID: 000184500120

Device Name	Directors				Device		
Physical	Sym	SA	:P	DA :IT	Config	Attribute	Sts Cap (MB)
/dev/rdisk/clt0d1s2	0029	04B:0	16B:C3	RDF2		N/Grp'd	WD 103
/dev/rdisk/clt0d2s2	0033	04B:0	15A:C3	RDF1		N/Grp'd	RW 103
/dev/rdisk/clt0d3s2	003D	04B:0	02B:D2	Unprotected		N/Grp'd	RW 3

This `symcfg list -address -available` command displays the VBUS, TID, and LUN addresses associated with front-end director 04B, port 0, and indicates the next available LUN address (generated by adding 1 to the last LUN address used). To decide on a LUN value, consider the LUN conventions for your host platform. The example uses LUN 00B as the address for the first new device:

```
symcfg list -sid 120 -sa 04B -p 0 -address -available
```

Symmetrix ID: 000184500120

Director			Device Name			Attr	Address		
Ident	Symbolic	Port	Sym	Physical			VBUS	TID	LUN
FA-4B	04B	0	-	AVAILABLE			0	0	000 *
			0029	/dev/rdisk/clt0d1s2			0	0	001
			0033	/dev/rdisk/clt0d2s2			0	0	002
			003D	/dev/rdisk/clt0d3s2			0	0	003
			0046	Not Visible			0	0	004
			-	AVAILABLE			0	0	005 *
			0075	Not Visible			0	0	00A
			-	AVAILABLE			0	0	00B *

Legend for Available address:

(\*): The VBUS, TID, LUN address values represent a gap in the address assignments or are the next available address in the run

The `symconfigure verify` command verifies that the current Symmetrix configuration is available for host-initiated configuration changes (that is, a configuration change session can be opened):

```
# symconfigure -sid 120 verify
```

```
A Configuration Change Verification is in progress. Please wait...
  Establishing a configuration change session.....Established.
  Verifying configuration.....Verified.
  Terminating the configuration change session.....Done.
```

The configuration verification session has succeeded.

The following command uses the vi text editor to create a text file named `map_dev.cmd`. As was done here, you can enter into the file the `map dev` entries that define the director, port, and LUN addresses for the four new devices. If you are mapping to a fibre adapter (FA) port system and are *not* using volume set addressing (as is the case for this example), only the LUN address is required (not the VBUS or TID):

```
vi map_dev.cmd

map dev 09B to dir 04B:0, lun=00B;
map dev 09C to dir 04B:0, lun=00C;
map dev 09D to dir 04B:0, lun=00D;
map dev 09E to dir 04B:0, lun=00E;
```

The `symconfigure` command with the `commit` argument executes the command file and begins the process of mapping the devices:

```
symconfigure -sid 120 -file map_dev.cmd -v -noprompt commit
```

```
Establishing a monitoring session.....Established.
The session changes are in the class of: Mapping/unmapping devices
{
  map dev 009B to dir 4B:0, target=00, lun=0B;
  map dev 009C to dir 4B:0, target=00, lun=0C;
  map dev 009D to dir 4B:0, target=00, lun=0D;
  map dev 009E to dir 4B:0, target=00, lun=0E;
}

The last action requested was: PREPARE
The state of the last action is: Running
PREPARE.....Done.
The last action requested was: COMMIT
The state of the last action is: Running
Step 05 of 44 steps.....Executing.
.....
Step 43 of 44 steps.....Executing.
Monitored session has terminated
Terminating the monitoring session.....Done.
```

To monitor the configuration change session while the `symconfigure commit` operation is processing, issue the `symconfigure query` command (as shown in examples 3 and 4).

The `symdev list` command shows that devices 009B through 009E are now mapped to a Symmetrix front-end port (04B:0) but that the new devices are not yet visible to the host:

```
symdev list -sid 120
```

```
Symmetrix ID: 000184500120
```

Device Name		Directors		Device			
Sym	Physical	SA	:P DA	:IT Config	Attribute	Sts	Cap (MB)
0000	Not Visible	???:?	01A:C0	Unprotected	N/Grp'd	RW	103
0001	Not Visible	???:?	16A:D4	Unprotected	N/Grp'd	RW	103
0002	Not Visible	13A:0	02A:C0	Unprotected	N/Grp'd	RW	103
0003	Not Visible	13A:1	15A:D4	Unprotected	N/Grp'd	RW	103
0004	Not Visible	13B:0	15A:C0	Unprotected	N/Grp'd	RW	103
0005	Not Visible	13B:1	02A:D4	Unprotected	N/Grp'd	RW	103
0006	Not Visible	14A:0	16A:C0	Unprotected	N/Grp'd	RW	103
0007	Not Visible	14A:1	01A:D4	Unprotected	N/Grp'd	RW	103
0008	Not Visible	14B:0	01B:C0	Unprotected	N/Grp'd	RW	103



0009	Not Visible	14B:1	16B:C4	Unprotected	N/Grp'd	RW	103
000A	Not Visible	????:?	02B:C0	BCV	N/Asst'd	RW	103
000B	Not Visible	????:?	15B:C4	Unprotected	N/Grp'd	(M) RW	309
000C	Not Visible	????:?	15B:C0	Unprotected	N/Grp'd	(m) RW	-
000D	Not Visible	????:?	02B:C4	Unprotected	N/Grp'd	(m) RW	-
000E	Not Visible	????:?	16B:C0	Unprotected	N/Grp'd	RW	103
000F	Not Visible	????:?	01B:C4	Unprotected	N/Grp'd	RW	103
0010	Not Visible	????:?	01A:D0	Unprotected	N/Grp'd	(M) RW	516
0011	Not Visible	????:?	16A:C4	Unprotected	N/Grp'd	(m) RW	-
0012	Not Visible	????:?	02A:D0	Unprotected	N/Grp'd	(m) RW	-
0013	Not Visible	????:?	15A:C4	Unprotected	N/Grp'd	(m) RW	-
<hr/>							
0099	Not Visible	????:?	15B:C1	Unprotected	N/Grp'd	RW	103
009A	Not Visible	????:?	01A:C3	Unprotected	N/Grp'd	RW	103
009B	Not Visible	04B:0	01A:C5	2-Way Mir	N/Grp'd	RW	103
009C	Not Visible	04B:0	02B:C1	2-Way Mir	N/Grp'd	RW	103
009D	Not Visible	04B:0	16A:C4	2-Way Mir	N/Grp'd	RW	103
009E	Not Visible	04B:0	15B:C3	2-Way Mir	N/Grp'd	RW	103

- ◆ The `sympd list` command confirms that the new devices are not visible to the host. This command displays only those devices that have a physical device name, which means that host can “see” them:

**sympd list -sid 120**

Symmetrix ID: 000184500120

Device Name	Directors			Device			
Physical	Sym	SA	:P DA	:IT Config	Attribute	Sts	Cap (MB)
<hr/>							
/dev/rdisk/clt0d1s2	0029	04B:0	16B:C3	RDF2	N/Grp'd	WD	103
/dev/rdisk/clt0d2s2	0033	04B:0	15A:C3	RDF1	N/Grp'd	RW	103
/dev/rdisk/clt0d3s2	003D	04B:0	02B:D2	Unprotected	N/Grp'd	RW	3

Executing the following utilities introduces the new devices to the host in a Sun Solaris environment:

```
drvconfig
disks
devlinks
```

After performing the proper host procedures to update the host view, you need to complete host addressing by making sure that the host address is recognized in the SYMAPI view. To update the SYMAPI database on your host, issue the `symcfg discover` command:

**symcfg discover**

This operation may take up to a few minutes. Please be patient...

The `sympd list` command shows that the new devices are now visible to the host:

**sympd list -sid 120**

Symmetrix ID: 000184500120

Device Name	Directors			Device			
Physical	Sym	SA	:P DA	:IT Config	Attribute	Sts	Cap (MB)
<hr/>							

```

/dev/rdisk/clt0d1s2      0029 04B:0 16B:C3 RDF2          N/Grp'd      WD      103
/dev/rdisk/clt0d2s2      0033 04B:0 15A:C3 RDF1          N/Grp'd      RW      103
/dev/rdisk/clt0d3s2      003D 04B:0 02B:D2 Unprotected N/Grp'd      RW       3
/dev/rdisk/clt0d32s2     009B 04B:0 01A:C0 2-Way Mir       N/Grp'd      RW     103
/dev/rdisk/clt0d33s2     009C 04B:0 16A:D4 2-Way Mir       N/Grp'd      RW     103
/dev/rdisk/clt0d34s2     009D 04B:0 01A:D2 2-Way Mir       N/Grp'd      RW     103
/dev/rdisk/clt0d35s2     009E 04B:0 01A:D2 2-Way Mir       N/Grp'd      RW     103

```

## Setting front-end port attributes

The hardware setup for this example is a Solaris host (api145) connected to a Symmetrix array (sid 120). The host is connected to front-end director FA-4A, which is a Fibre Channel director. The example illustrates how to modify a front-end port attribute (SCSI\_3).

- ◆ Using the `-connections` option with `symcfg list` allows you to see the host connections to a Symmetrix array. The following display shows the front-end director (FA-4A) that this host uses to reach the Symmetrix (sid 120):

```
symcfg list -sid 120 -connections
```

```
Symmetrix ID:      000192601120
```

Symmetrix		Host				
Director	Port	Node Name	IP Address	HW Type	OS Name	OS Revision
UN-1A	0	HK192601135	0.0.0.1	INTEL	WinNT-SP	5.1.2600
FA-8E	0	laqa1043	10.242.51.43	x86_64	LINUX	2.6.18-164
FA-8E	0	laqa1041	*4675:834a:b9c7	AMD64	WinNT	6.0.6002
FA-8E	0	laqa0092	10.242.50.92	i86pc	SunOS	5.10
FA-8E	0	laqa1025	*2336:2ba9:79fe	x86_64	WinNT	6.1.7600
FA-8E	0	laqa0083	10.242.50.83	sun4u	SunOS	5.10
FA-8E	0	laqa1042	10.242.51.42	x86_64	LINUX	2.6.27.19-
FA-8E	0	laqa0038	10.242.50.38	i86pc	SunOS	5.10
FA-8E	0	laqa0240	10.242.50.240	00CB718	AIX	6.1
FA-8E	0	laqa0124	10.242.50.124	i86pc	SunOS	5.10

- ◆ The Symmetrix port attribute `VCM_State` is a fibre protocol flag that can be either enabled or disabled (the default). You need to enable this flag for device masking or Volume Logix software (which provides volume management controls to handle access to Symmetrix devices). The `symcfg list` command provides detailed (`-v`) information about the Symmetrix configuration and the front-end director/port that connects the host to the Symmetrix array. The section *Fibre Specific Flags*, in the example below, shows that `VCM_State` is currently disabled. The ellipsis (.....) indicates where some output was omitted for brevity:

```
symcfg list -sid 120 -sa 04A -p 0 -v
```

```
Symmetrix ID: 000184500120
```

```

Product Model           : 8430
Symmetrix ID            : 000184500120

Microcode Version (Number) : 5568
Microcode Date          : 11.12.2001

Microcode Patch Date     : 11.12.2001
Microcode Patch Level    : 37
.....

```

```

Number of Configured (Sym) Devices : 159
Number of Visible (Host) Devices  : 6
Number of Configured Actual Disks : 96
Number of Configured Hot Spares   : 0

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  : 8FB66
Switched RDF Configuration State  : Disabled
Concurrent RDF Configuration State : Disabled
Dynamic RDF Configuration State   : Disabled
RDF Data Mobility Configuration State: Disabled
Access Control Configuration State : Disabled
Device Masking (VCM) Config State : Disabled

```

#### Director Identification: FA-4A

```

Director Type           : FibreChannel
Director Status         : Online

Number of Director Ports : 1
Director Ports Status    : [ON,N/A,N/A,N/A]

Director Symbolic Number : 04A
Director Numeric Number  : 4
Director Slot Number     : 4

```

#### Director Port: 0

```

WWN Node Name           : 50060482bfcfe603
WWN Port Name           : 50060482bfcfe603

```

```

Fibre Channel Loop ID   : 0
Fibre Adapter Type      : N/A

```

```

SCSI Flags
{
    Tagged_Commands      : Enabled
    Linked_Commands      : Disabled
    Sync_Transfer        : Disabled
    Wide_Transfer        : Disabled
    Negotiate_Reset      : Disabled
    Soft_Reset           : Disabled
    Environ_Set          : Disabled
    Cyl_Count_In_Name    : Disabled

```

```

}
```

```

Fibre Specific Flags
{
    Disk_Array           : Disabled
    Volume_Set_Addressng : Disabled
    Hard_Addressng       : Enabled
    Non_Participating    : Disabled
    Global_3rdParty_Logout : Disabled
    Init_Point_to_Point  : Disabled
    Unique_WWN           : Enabled
    Generic_VSA          : Disabled
    VCM_State            : Disabled
    Class_2_Service      : Disabled

```

```

    OpenVMS                               : Disabled
}

```

- ◆ The `symconfigure verify` command verifies that the current Symmetrix configuration is available for host-initiated configuration changes (that is, a configuration change session can be opened):

```
symconfigure -sid 120 verify
```

```

A Configuration Change Verification is in progress. Please wait...
  Establishing a configuration change
  session.....Established.
    Verifying
  configuration.....Verified.
    Terminating the configuration change session.....Done.

The configuration verification session has succeeded.

```

- ◆ The `symconfigure` command with the `commit` argument executes the command file and begins the process of setting the SCSI\_3 flag to “disable:”

```
symconfigure -sid 397 -cmd "set port 07f:1 SCSI_3=disable;" commit -v -noprompt
```

```
A Configuration Change operation is in progress. Please wait...
```

```

  Establishing a configuration change session.....Established.
  Processing symmetrix 000194900397
  {
    set port 8F:0 SCSI_3(SC3)=Disable;
  }

```

```

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 004 of 046 steps.....Executing.
Step 017 of 046 steps.....Executing.
Step 023 of 046 steps.....Executing.
Step 025 of 079 steps.....Executing.
Step 041 of 079 steps.....Executing.
Step 055 of 079 steps.....Executing.
Step 055 of 079 steps.....Executing.
Step 060 of 079 steps.....Executing.
Step 061 of 079 steps.....Executing.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.

```

```
The configuration change session has successfully completed.
```

- ◆ To monitor the configuration change session while the `symconfigure commit` operation is processing, you can issue the `symconfigure query` command or the following UNIX `tail` command from a second window. The following two commands allow you to compare outputs from the `tail` command and from `symconfigure query`:

```
tail -f /var/symapi/log/symapi-20011228.log
```

```

12/28/2001 16:10:34.404 22122 0 EMC:SYMCLI iCfgChgSessionStart Called to start a local
session for SID 000184500120
12/28/2001 16:10:40.524 22122 Establishing session with Local cfg srvr
(000184500120)...Established.
12/28/2001 16:10:47.871 22122 {
12/28/2001 16:10:47.878 22122   set port 4A:0 VCM_State=ENABLE;
12/28/2001 16:10:47.881 22122 }

```

```

12/28/2001 16:10:52.891 22122 Submitting configuration changes.....Submitted.
12/28/2001 16:11:03.901 22122 Validating configuration changes.....Validated.
12/28/2001 16:11:14.922 22122 Initiating PREPARE of configuration changes..Queued.
12/28/2001 16:11:26.922 22122 PREPARE requesting required resources.....Obtained.
12/28/2001 16:11:28.932 22122 Step 004 of 011 steps.....Executing.
12/28/2001 16:11:34.942 22122 Step 007 of 011 steps.....Executing.
12/28/2001 16:11:41.952 22122 Step 008 of 011 steps.....Executing.
12/28/2001 16:12:03.972 22122 Local: PREPARE.....Done.
12/28/2001 16:12:15.082 22122 Initiating COMMIT of configuration changes...Queued.
12/28/2001 16:12:27.083 22122 COMMIT requesting required resources.....Obtained.
12/28/2001 16:12:28.093 22122 Step 004 of 039 steps.....Executing.
.....
12/28/2001 16:17:13.577 22122 Step 038 of 039 steps.....Executing.
12/28/2001 16:17:19.587 22122 Local: COMMIT.....Done.
12/28/2001 16:17:19.602 22122 0 EMC:SYMCLI process_load_request Switching to FULL load for
000184500120 because the configuration changed
12/28/2001 16:17:26.275 22122 Terminating session with configuration server..Done.

```

- ◆ The `symconfigure query` command issued from a second window checks the status of the configuration change session 30 times at 10-second intervals:

```
symconfigure query -sid 120 -i 10 -c 30
```

```

Establishing a monitoring session.....Established.
The session changes are in the class of: Modifying port configurations
{
    set port 4A:0
        VCM_State=enable;
}

The last action requested was: PREPARE
The state of the last action is: Running
PREPARE.....Done.
The last action requested was: COMMIT
The state of the last action is: Running
Step 06 of 39 steps.....Executing.
.....
Step 38 of 39 steps.....Executing.
COMMIT.....Done.
Monitored session has terminated
Terminating the monitoring session.....Done.

```

- ◆ The `symcfg list` command again provides the detailed (-v) information about the Symmetrix configuration and the front-end director/port that connects the host to the Symmetrix array. The section Fibre Specific Flags, in the example below, shows that VCM\_State is now enabled:

```
symcfg -sa 04A -p 0 list -v -sid 120
```

```
Symmetrix ID: 000184500120
```

```

Product Model                : 8430
Symmetrix ID                  : 000184500120

Microcode Version (Number)    : 5568 (15BFAA01)
Microcode Date                 : 11.12.2001

Microcode Patch Date          : 11.12.2001
Microcode Patch Level         : 37

```

```
.....
Director Identification: FA-4A
```

```

Director Type                 : FibreChannel
Director Status                : Online

```

```

Number of Director Ports          : 1
Director Ports Status             : [ON,N/A,N/A,N/A]

Director Symbolic Number          : 04A
Director Numeric Number           : 4
Director Slot Number              : 4

Director Port: 0

WWN Node Name                     : 50060482bfcfe603
WWN Port Name                     : 50060482bfcfe603

Fibre Channel Loop ID            : 0
Fibre Adapter Type               : N/A

```

---

```

Fibre Specific Flags
{
    Disk_Array                  : Disabled
    Volume_Set_Addresssing     : Disabled
    Hard_Addresssing           : Enabled
    Non_Participating          : Disabled
    Global_3rdParty_Logout     : Disabled
    Init_Point_to_Point        : Disabled
    Unique_WWN                 : Enabled
    Generic_VSA                : Disabled
    VCM_State                  : Enabled
    Class_2_Service            : Disabled
    OpenVMS                    : Disabled
}

```

## Configuring concurrent SRDF devices

This example is performed using Solutions Enabler version 6.0. You can use the SYMCLI Configuration Manager to configure concurrent dynamic SRDF pairs while the Symmetrix array is running. In this Fibre Channel example, the controlling host is connected to a local Symmetrix array (sid 011). The local Symmetrix array is connected via SRDF links to two remote Symmetrix arrays (sid 093 and sid 201). The example uses two different SRDF (RA) groups to achieve the connection between each local R1 device and its two remote R2 mirrors.

For brevity, this example shows creating an SRDF group with only one connection (one local director to one remote director). Using recommended practice, you would have at least two sets of RA directors supporting the SRDF group.

- ◆ The following command uses the vi text editor to create a text file named `set_sym.cmd`. As shown here, you can enter into this file the `set symmetrix` entries that set the three flags necessary to enable concurrent dynamic SRDF in the Symmetrix array attached to your host (these flags may or may not have been set using SymmWin). You will also need to ensure that the `dynamic_rdf` flag is enabled in the two remote Symmetrix arrays (enabling the concurrent flags in the remote arrays is not necessary).

```

vi set_sym.cmd

set symmetrix dynamic_rdf=enable;
set symmetrix concurrent_dynamic_rdf=enable;
set symmetrix concurrent_rdf=enable;

```

- ◆ The `symconfigure commit` command executes the command file and initiates the process of setting the Symmetrix `dynamic_rdf` parameter and the `concurrent_dynamic_rdf` flag. The ellipsis (.....) indicates output that was omitted for brevity:

```
symconfigure -sid 11 -file set_sym.cmd -v -noprompt commit
```

```
Establishing a configuration change session.....Established.
Processing symmetrix 000187400011
{
  set symmetrix dynamic_rdf_configuration = Enabled;
  set symmetrix concurrent_dynamic_rdf = Enabled;
  set symmetrix concurrent_rdf = Enabled;
}

Submitting configuration changes.....Submitted.
Validating configuration changes.....Validated.
Initiating PREPARE of configuration changes.....Queued.
PREPARE requesting required resources.....Obtained.
Step 008 of 011 steps.....Executing.
Local: PREPARE.....Done.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 012 of 034 steps.....Executing.
.....
Step 032 of 034 steps.....Executing.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.
```

- ◆ The `symcfg list` command displays current SRDF (RA) groups that serve as SRDF links to connect local Symmetrix 011 to remote Symmetrix 093 through director 4D:

```
symcfg list -ra 4D -sid 11
```

```
Symmetrix ID: 000187400011
```

S Y M M E T R I X					R D F		D I R E C T O R S			
Ident	Symb	Num	Slot	Type	Attr	Remote SymmID	Local RA Grp	Remote RA Grp	Status	
RF-4D	04D	52	4	RDF-R1	-	000187400093	27 (1A)	23 (16)	Online	
					-	000187400093	28 (1B)	24 (17)		
					-	000187400093	29 (1C)	25 (18)		
					-	000187400093	30 (1D)	26 (19)		
					-	000187400093	31 (1E)	27 (1A)		

- ◆ The following `symrdf addgrp` command creates a dynamic SRDF group that represents another SRDF link between Symmetrix 000187400011 and Symmetrix 000187400093. It adds dynamic SRDF group 58 on the local Symmetrix array, and SRDF group 58 on the remote Symmetrix array. You must specify a group label (`grp58` in this case) that can be used when modifying or deleting the group. Creation of the dynamic SRDF group includes director 4D from the local Symmetrix and 3C from the remote Symmetrix as the director end points of this connection.

It is important to be aware of your network topology when creating dynamic SRDF groups between two Symmetrix arrays. To create a dynamic SRDF link (a connection) between RA directors, the director end points must be able to see each other through the Fibre Channel fabric. For example, a dynamic SRDF link can be created between local and remote directors only if the Fibre Channel zoning is set up so that the two directors can see each other through the fabric:

```
symrdf -v addgrp -label grp58 -rdfg 58 -sid 187400011 -dir 4D  
-remote_rdfg 58 -remote_sid 000187400093 -remote_dir 3C -noprompt
```

An RDF Addgrp operation execution is in progress for dynamic group 'grp58'. Please wait...

Successfully Added Dynamic RDF Group 'grp58' for Symm: 000187400011

---

**Note:** For brevity, this command creates an SRDF group with only one connection (local director 4D to remote director 3C). Using recommended practice, you would have at least two sets of RA directors supporting the SRDF group. For example, the previous command could include `-dir 4D,4C` and `-remote_dir 3C,3D`.

---

- ◆ Another `symcfg list` command verifies the logical connections from the local director (4D) point of view. Dynamic SRDF group 58 has been added to both the local and remote Symmetrix arrays:

```
symcfg list -ra 4D -sid 11
```

Symmetrix ID: 000187400011

S Y M M E T R I X     R D F     D I R E C T O R S										
Ident	Symb	Num	Slot	Type	Attr	Remote SymmID	Local RA Grp	Remote RA Grp	Status	
RF-4D	04D	52	4	RDF-R1	-	000187400093	27 (1A)	23 (16)	Online	
					-	000187400093	28 (1B)	24 (17)		
					-	000187400093	29 (1C)	25 (18)		
					-	000187400093	30 (1D)	26 (19)		
					-	000187400093	31 (1E)	27 (1A)		
					-	000187400093	58 (39)	58 (39)		

- ◆ Another `symcfg list` displays current SRDF (RA) groups that serve as SRDF links to connect local Symmetrix 000187400011 to the second remote Symmetrix (000000006201) through director 13A:

```
symcfg list -ra 13A -sid 11
```

Symmetrix ID: 000187400011

S Y M M E T R I X     R D F     D I R E C T O R S										
Ident	Symb	Num	Slot	Type	Attr	Remote SymmID	Local RA Grp	Remote RA Grp	Status	
RE-13A	13A	13	13	RDF-R1	-	000000006201	47 (2E)	47 (2E)	Online	
					-	000000006201	50 (31)	50 (31)		

- ◆ The following `symrdf addgrp` command creates a dynamic SRDF group that represents another SRDF link between Symmetrix 000187400011 and Symmetrix 000000006201. It adds dynamic SRDF group 51 on the local Symmetrix, and SRDF group 51 on the remote Symmetrix. Creation of the local and remote SRDF groups includes director 13A from the local Symmetrix and 13B from the remote Symmetrix array:

```
symrdf -v addgrp -label grp51 -rdfg 51 -sid 187400011 -dir 13A  
-remote_rdfg 51 -remote_sid 000000006201 -remote_dir 13B -noprompt
```



An RDF Addgrp operation execution is in progress for dynamic group 'grp51'. Please wait...

Successfully Added Dynamic RDF Group 'grp51' for Symm: 000187400011

**Note:** For brevity, this command creates an SRDF group with only one connection (local director 13A to remote director 13B). Using recommended practice, you would have at least two sets of RA directors supporting the SRDF group. For example, the previous command could include `-dir 13A,13B` and `-remote_dir 13B,13C`.

- ◆ Another `symcfg list` command verifies the logical connections from the local director (13A) point of view. Dynamic SRDF group 51 has been added to both the local and remote Symmetrix arrays:

```
symcfg list -ra 13A -sid 11
```

Symmetrix ID: 000187400011

S Y M M E T R I X     R D F     D I R E C T O R S										
Ident	Symb	Num	Slot	Type	Attr	Remote SymmID	Local RA Grp	Remote RA Grp	Status	
RE-13A	13A	13	13	RDF-R1	-	000000006201	47 (2E)	47 (2E)	Online	
					-	000000006201	50 (31)	50 (31)		
					-	000000006201	51 (32)	51 (32)		

The `symdev list` command with the `-dynamic` option displays those devices on the local Symmetrix arrays that have been configured to be capable of dynamic RDF. These will be the source devices:

```
symdev list -dynamic -sid 11
```

Symmetrix ID: 000187400011

Device Name		Directors		Device			
Sym	Physical	SA :P	DA :IT	Config	Attribute	Sts	Cap (MB)
0300	/dev/vx/rdmp/c6t0d0s2	14C:0	16A:D8	RAID-5	N/Grp'd	RW	449
0301	/dev/vx/rdmp/c6t0d1s2	14C:0	02A:D8	RAID-5	N/Grp'd	RW	449
0302	/dev/vx/rdmp/c6t0d2s2	14C:0	01A:C4	RAID-5	N/Grp'd	RW	449
0303	/dev/vx/rdmp/c6t0d3s2	14C:0	15A:C0	RAID-5	N/Grp'd	RW	449
0304	/dev/vx/rdmp/c6t0d4s2	14C:0	16A:C3	RAID-5	N/Grp'd	RW	449
0305	/dev/vx/rdmp/c6t0d5s2	14C:0	02A:C3	RAID-5	N/Grp'd	RW	449

- ◆ The following command uses the vi text editor to create a text file named `OEA2OEB2.list`. As was done here, you can enter into the file those device names that will constitute one of the sets of dynamic SRDF pairs (those R1/R2 pairs for the local Symmetrix and remote Symmetrix 000187400093). The R1 devices are listed in first column, and the remote R2 devices are listed in the second column on the same line as their respective R1 source. Like the R1 devices, the R2 devices must also be non-RDF devices that have been set with the dynamic SRDF attribute:

```
vi OEA2OEB2.list
```

```
300 080
301 081
```

```
302 082
303 083
304 084
305 085
```

- ◆ The following command uses the vi text editor to create a text file named `OEA206A2.list`. As was done here, you can enter into the file those device names that will constitute the second set of dynamic SRDF pairs (those R1/R2 pairs for the local Symmetrix and remote Symmetrix 000000006201):

```
vi OEA206A2.list
```

```
300 150
301 151
302 152
303 153
304 154
305 155
```

- ◆ The `symrdf createpair` command parses the file called `OEA20EB2.list` that defines the dynamic SRDF pairs and specifies that the column-1 devices in the file are R1 devices (`-type RDF1`) on the local Symmetrix (000187400011). Communication is done via SRDF group 58 (`-rdfg 58`), which was previously established as the SRDF link to remote Symmetrix 000187400093:

```
symrdf createpair -file OEA20EB2.list -sid 11 -rdfg 58 -type rdf1 -establish -noprompt
```

An RDF 'Create Pair' operation execution is in progress for device file 'OEA20EB2.list'. Please wait...

```
Create RDF Pair in (0011,58).....Done.
Mark target device(s) in (0011,58) for full copy from source....Started.
Devices: 0300-0305 ..... Marked.
Mark target device(s) in (0011,58) for full copy from source....Done.
Merge track tables between source and target in (0011,58).....Started.
Devices: 0300-0305 ..... Merged.
Merge track tables between source and target in (0011,58).....Done.
Resume RDF link(s) for device(s) in (0011,58).....Started.
Resume RDF link(s) for device(s) in (0011,58).....Done.
```

The RDF 'Create Pair' operation successfully executed for device file 'OEA20EB2.list'.

- ◆ A second `symrdf createpair` command parses the file called `OEA206A2.list` that defines the dynamic SRDF pairs and specifies that the column-1 devices in the file are R1 devices (`-type RDF1`) on the local Symmetrix (000187400011). Communication is done via SRDF group 51 (`-rdfg 51`), which was previously established as the SRDF link to remote Symmetrix 000000006201:

```
symrdf createpair -file OEA206A2.list -sid 11 -rdfg 51 -type rdf1 -establish -nop
```

An RDF 'Create Pair' operation execution is in progress for device file 'OEA206A2.list'. Please wait...

```
Create RDF Pair in (0011,51).....Done.
Mark target device(s) in (0011,51) for full copy from source....Started.
Devices: 0300-0305 ..... Marked.
Mark target device(s) in (0011,51) for full copy from source....Done.
Merge track tables between source and target in (0011,51).....Started.
Devices: 0300-0305 ..... Merged.
Merge track tables between source and target in (0011,51).....Done.
Resume RDF link(s) for device(s) in (0011,51).....Started.
```

Resume RDF link(s) for device(s) in (0011,51).....Done.

The RDF 'Create Pair' operation successfully executed for device  
file 'OEA206A2.list'.

- ◆ The `symdmg` command creates an RDF1 type device group named `dynConc`. The `symld` command adds to the group on which the R1 devices that were created, the local Symmetrix (sid 11):

```
symdmg create -type rdf1 dynConc  
symld -g dynConc -sid 11 -range 300:305 addall dev
```

- ◆ 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 concurrent pairs are in the Synchronized state:

```
symrdf -g dynConc query -rdfg all
```

```
Device Group (DG) Name      : dynConc  
DG's Type                   : RDF1  
DG's Symmetrix ID           : 000187400011  
Remote Symmetrix ID         : 000000006201  
RDF (RA) Group Number       : 51 (32)  
Remote Symmetrix ID         : 000187400093  
RDF (RA) Group Number       : 58 (39)
```

Source (R1) View					Target (R2) View					MODES	
-----					-----					-----	
Standard	ST				LI	ST					
Logical	A				N	A					
Device	T	R1 Inv	R2 Inv		K	T	R1 Inv	R2 Inv		RDF Pair	
Dev	E	Tracks	Tracks		S	Dev	E	Tracks	Tracks	MDA	STATE
-----					-----					-----	
DEV001	0300	RW	0	0	RW	0080	WD	0	0	S..	Synchronized
		RW	0	0	RW	0150	WD	0	0	S..	Synchronized
DEV002	0301	RW	0	0	RW	0081	WD	0	0	S..	Synchronized
		RW	0	0	RW	0151	WD	0	0	S..	Synchronized
DEV003	0302	RW	0	0	RW	0082	WD	0	0	S..	Synchronized
		RW	0	0	RW	0152	WD	0	0	S..	Synchronized
DEV004	0303	RW	0	0	RW	0083	WD	0	0	S..	Synchronized
		RW	0	0	RW	0153	WD	0	0	S..	Synchronized
DEV005	0304	RW	0	0	RW	0084	WD	0	0	S..	Synchronized
		RW	0	0	RW	0154	WD	0	0	S..	Synchronized
DEV006	0305	RW	0	0	RW	0085	WD	0	0	S..	Synchronized
		RW	0	0	RW	0155	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
```

## Updating a disk label with emulated device geometry

**Note:** Because changes in geometry are destructive, disk geometry changes should be made to a device prior to it being allocated to a host. If a device needs to be re-purposed and does not contain useful data, the device must be unmapped or unmasked and write disabled or not ready before its geometry can be modified.

The following procedure can be run to prepare the host volumes and update the disk label using only native operating system utilities:

1. Have an EMC Customer Engineer enable FBA Geometry Emulation or use Solutions Enabler to set the device geometry to SYMM-7.
2. Synchronize the source and target volumes.
3. Split the target volumes and insure that they are ready and read/write.
4. Print the VTOC of the source volumes to a file, as follows:

```
prtvtoc /dev/rdisk/SourceVolume > FileName
```

5. Transfer the VTOC files to the target host.
6. Update the manufacturer information on the target volumes, as follows:

```
format TargetVolume
type
Auto configure
label
```

7. Update the partition information on the target volumes, as follows:

```
fmthard -s FileName TargetVolume
```

### Procedure example

The following example shows the commands and the resulting output of a data migration and update to the disk label.

This example uses Solutions Enabler to modify the target disk geometry and SRDF to perform the migration. The disk group that will be migrated is called `geodg` and contains Symmetrix devices `036` and `037` from the source array which is serial number `0073`, and is a DMX 2000S-M2 running Enginuity 5671:

```
symcfg list
```

S Y M M E T R I X

SymmID	Attachment	Model	Mcode Version	Cache Size (MB)	Num Phys Devices	Num Symm Devices
000387720073	Local	2000S-M2	5671	65536	30	80
000190100414	Remote	DMX3-24	5773	131072	0	144

```
syminq | grep c2t50060485C708E268d11s2
```

/dev/rdisk/*d11s2	R1	EMC	SYMMETRIX	5671	7300036000	4417920
/dev/vx/rdmp/*d11s2	R1	EMC	SYMMETRIX	5671	7300036000	4417920

```
syminq | grep c2t50060485C708E268d12s2
```

```
/dev/rdsd/*d12s2    R1      EMC      SYMMETRIX    5671 7300037000    4417920
/dev/vx/rdmp/*d12s2 R1      EMC      SYMMETRIX    5671 7300037000    4417920
```

```
vxprint -htQq
dg geodg          default      default    29000      1203016756.36.licoa034

dm geodg01        c2t50060485C708E268d11s2 auto 2048 8831616 -
dm geodg02        c2t50060485C708E268d12s2 auto 2048 8831616 -

v  geodgvol01     -              ENABLED    ACTIVE    17663232 ROUND    -          fsgen
pl geodgplex01    geodgvol01    ENABLED    ACTIVE    17663232 CONCAT   -          RW
sd geodg01sd01    geodgplex01    geodg01    0          8831616  0          c2t5006*  ENA
sd geodg02sd01    geodgplex01    geodg02    0          8831616  8831616    c2t5006*  ENA
```

The disk group contains a single Veritas volume that is mounted and contains data:

```
df -k
```

Filesystem	kbytes	used	avail	capacity	Mounted on
/dev/dsk/c0t1d0s0	16379106	6176410	10038905	39%	/
/proc	0	0	0	0%	/proc
mnttab	0	0	0	0%	/etc/mnttab
fd	0	0	0	0%	/dev/fd
swap	1430576	112	1430464	1%	/var/run
dmpfs	1430464	0	1430464	0%	/dev/vx/dmp
dmpfs	1430464	0	1430464	0%	/dev/vx/rdmp
swap	1430840	376	1430464	1%	/tmp
/dev/vx/dsk/geodg/geodgvol01	8831616	4213799	4329252	50%	/mp/geodgvol01_mp

```
ls -l /mp/geodgvol01_mp
```

```
total 8388678
drwxr-xr-x  2 root    other          3072 Feb 14 16:36 flat_files
drwxr-xr-x  2 root    root              96 Feb 14 14:26 lost+found
-rw-----T  1 root    other    1073741824 Feb 14 16:37 testfile1
-rw-----T  1 root    other    1073741824 Feb 14 16:38 testfile2
-rw-----T  1 root    other    1073741824 Feb 14 16:39 testfile3
-rw-----T  1 root    other    1073741824 Feb 14 16:41 testfile4
```

```
ls -l /mp/geodgvol01_mp/flat_files
```

```
total 290
-rw-r--r--  1 root    other          131 Feb 14 16:36 arp.conf
-rw-r--r--  1 root    other         1583 Feb 14 16:36 audiocs.conf
-rw-r--r--  1 root    other         1414 Feb 14 16:36 audioens.conf
-rw-r--r--  1 root    other         1601 Feb 14 16:36 audiotests.conf
-rw-r--r--  1 root    other         1810 Feb 14 16:36 bofi.conf
-rw-r--r--  1 root    other          135 Feb 14 16:36 clone.conf
-rw-r--r--  1 root    other          129 Feb 14 16:36 cn.conf
-rw-r--r--  1 root    other          139 Feb 14 16:36 conskbd.conf
-rw-r--r--  1 root    other          164 Feb 14 16:36 consms.conf
```

```
cat /mp/geodgvol01_mp/flat_files/arp.conf
#
# Copyright (c) 1992, by Sun Microsystems, Inc.
#
#ident  "@(#)arp.conf  1.4      93/06/03 SMI"
name="arp" parent="pseudo" instance=0;
```

The targets of the migration in Symmetrix 0414 (DMX3-24, Enginuity 5773) are devices 076 and 077. Before migrating, the device geometry should be correctly set to SYMM-6 so that they match the source devices on the DMX:

```
symconfigure -sid 14 -cmd "set dev 0076:0077 geometry=SYMM-6 cyls=9204;" commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000190100414
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Locking devices.....Locked.
Validating configuration changes.....Validated.
Preparing configuration changes.....Prepared.
Committing configuration changes.....Started.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

```
symdev list -geometry_set -sid 14
```

Symmetrix ID: 000190100414

Device Name		Directors		Device			
Sym	Physical	SA	:P DA :IT	Config	Attribute	Sts	Cap (MB)
0076	Not Visible	???:?	16C:C9	2-Way Mir	N/Grp'd	RW	4314
0077	Not Visible	???:?	16D:DB	2-Way Mir	N/Grp'd	RW	4314

**Note:** Setting the device geometry can be performed before or after the devices are configured as SRDF devices.

The devices are configured as SRDF devices using dynamic SRDF (RDF group 01 already exists), and is synchronized:

```
cat dynamic
036 076
037 077
```

```
symrdf -sid 73 -file dynamic -rdfg 1 createpair -type R1 -establish -nop
```

An RDF 'Create Pair' operation execution is in progress for device file 'dynamic'. Please wait...

```
Create RDF Pair in (0073,001).....Started.
Create RDF Pair in (0073,001).....Done.
Mark target device(s) in (0073,001) for full copy from source....Started.
Devices: 0036-0037 in (0073,001).....Marked.
Mark target device(s) in (0073,001) for full copy from source....Done.
Merge track tables between source and target in (0073,001).....Started.
Devices: 0036-0037 in (0073,001).....Merged.
Merge track tables between source and target in (0073,001).....Done.
Resume RDF link(s) for device(s) in (0073,001).....Started.
Resume RDF link(s) for device(s) in (0073,001).....Done.
```

The RDF 'Create Pair' operation successfully executed for device file 'dynamic'.

A Symmetrix device group containing the two devices has been created and the devices have been synchronized:

**symrdf -g vxgeodg query**

```
Device Group (DG) Name      : vxgeodg
DG's Type                  : RDF1
DG's Symmetrix ID          : 000387720073 (Microcode Version: 5671)
Remote Symmetrix ID        : 000190100414 (Microcode Version: 5773)
RDF (RA) Group Number     : 1 (00)
```

Source (R1) View					Target (R2) View					MODES	
Standard	ST				LI	ST					
Logical	A				N	A					
Device	T	R1 Inv	R2 Inv	K	T	R1 Inv	R2 Inv			RDF Pair	
Dev	E	Tracks	Tracks	S	Dev	E	Tracks	Tracks	MDA	STATE	
DEV001	0036	RW	0	0	RW	0076	WD	0	0	S..	Synchronized
DEV002	0037	RW	0	0	RW	0077	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
```

A copy of the VTOC from the source volumes must be saved so that it can be written to the target devices after they are labeled. This is done by saving the output of prtvtoc to a file:

```
prtvtoc /dev/rdisk/c3t50060485C708E247d11s2 > /mig_files/d11_0036.vtoc
```

```
prtvtoc /dev/rdisk/c3t50060485C708E247d12s2 > /mig_files/d12_0037.vtoc
```

```
cat /mig_files/d11_0036.vtoc
* /dev/rdisk/c3t50060485C708E247d11s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   64 sectors/track
*   15 tracks/cylinder
*   960 sectors/cylinder
*   9204 cylinders
*   9202 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
* Partition Tag Flags First Sector Sector Count Last Sector Mount Directory
*   2      5   01      0    8833920    8833919
*   7     15   01      0    8833920    8833919
```

```
# cat /mig_files/d12_0037.vtoc
* /dev/rdisk/c3t50060485C708E247d12s2 partition map
*
```

```

* Dimensions:
*   512 bytes/sector
*   64 sectors/track
*   15 tracks/cylinder
*   960 sectors/cylinder
*   9204 cylinders
*   9202 accessible cylinders
*
* Flags:
*   1: unmountable
*  10: read-only
*
*
* Partition  Tag  Flags      First      Sector      Last
*           Tag  Flags      Sector      Count      Sector  Mount Directory
*           2    5    01          0    8833920    8833919
*           7   15    01          0    8833920    8833919

```

The VTOC files from the source volumes can then be transferred to the target host. The SRDF pairs can now be split:

```
symrdf -g vxgeodg split -nop
```

An RDF 'Split' operation execution is in progress for device group 'vxgeodg'. 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 'vxgeodg'.

The target devices are now available to the target host:

```

./inq -no_dots | grep 0076

/dev/rdisk/c2t3d11s2      :EMC      :SYMMETRIX      :5773  :1400076000 :      4417920
/dev/vx/rdmp/c2t3d11s2  :EMC      :SYMMETRIX      :5773  :1400076000 :      4417920

./inq -no_dots | grep 0077

/dev/rdisk/c2t3d12s2      :EMC      :SYMMETRIX      :5773  :1400077000 :      4417920
/dev/vx/rdmp/c2t3d12s2  :EMC      :SYMMETRIX      :5773  :1400077000 :      4417920

```

They contain the label from the source DMX. The labels can now be rewritten on the target devices:

```

...
47. c2t3d11 <EMC-SYMMETRIX-5671 cyl 9202 alt 2 hd 15 sec 64>
    /pci@1f,4000/fibre-channel@5/sd@3,b
48. c2t3d12 <EMC-SYMMETRIX-5671 cyl 9202 alt 2 hd 15 sec 64>
    /pci@1f,4000/fibre-channel@5/sd@3,c

```

```

Specify disk (enter its number): 47
selecting c2t3d11
[disk formatted]

```

```

FORMAT MENU:
disk          - select a disk
type          - select (define) a disk type
partition     - select (define) a partition table
current       - describe the current disk
format        - format and analyze the disk
repair        - repair a defective sector
label         - write label to the disk
analyze       - surface analysis
defect        - defect list management

```



```

        backup      - search for backup labels
        verify      - read and display labels
        save         - save new disk/partition definitions
        inquiry      - show vendor, product and revision
        volname      - set 8-character volume name
        !<cmd>       - execute <cmd>, then return
        quit
format> type

```

## AVAILABLE DRIVE TYPES:

0. Auto configure
1. Quantum ProDrive 80S
2. Quantum ProDrive 105S
3. CDC Wren IV 94171-344
4. SUN0104
5. SUN0207
6. SUN0327
7. SUN0340
8. SUN0424
9. SUN0535
10. SUN0669
11. SUN1.0G
12. SUN1.05
13. SUN1.3G
14. SUN2.1G
15. SUN2.9G
16. Zip 100
17. Zip 250
18. SUN36G
19. EMC-SYMMETRIX-5773
20. EMC-SYMMETRIX-5773
21. EMC-SYMMETRIX-5773
22. EMC-SYMMETRIX-5671
23. other

```

Specify disk type (enter its number)[22]: 0
c2t3d11: configured with capacity of 4.21GB
<EMC-SYMMETRIX-5773 cyl 9202 alt 2 hd 15 sec 64>
selecting c2t3d11
[disk formatted]
format> label
Ready to label disk, continue? Y

```

```

Specify disk (enter its number)[48]: 48
selecting c2t3d12
[disk formatted]
format> type

```

## AVAILABLE DRIVE TYPES:

0. Auto configure
1. Quantum ProDrive 80S
2. Quantum ProDrive 105S
3. CDC Wren IV 94171-344
4. SUN0104
5. SUN0207
6. SUN0327
7. SUN0340
8. SUN0424
9. SUN0535
10. SUN0669
11. SUN1.0G
12. SUN1.05
13. SUN1.3G
14. SUN2.1G

```

15. SUN2.9G
16. Zip 100
17. Zip 250
18. SUN36G
19. EMC-SYMMETRIX-5773
20. EMC-SYMMETRIX-5773
21. EMC-SYMMETRIX-5773
22. EMC-SYMMETRIX-5671
23. EMC-SYMMETRIX-5773
24. other
Specify disk type (enter its number)[22]: 0
c2t3d12: configured with capacity of 4.21GB
<EMC-SYMMETRIX-5773 cyl 9202 alt 2 hd 15 sec 64>
selecting c2t3d12
[disk formatted]
format> label
Ready to label disk, continue? y

format> q
#

```

The target device labels now contain the correct Manufacturer ID and disk geometry. The VTOC that was copied over from the source host can now be written to the target devices:

```

fmthard -s /mig_files/d11_0036.vtoc /dev/rdisk/c2t3d11s2

fmthard: New volume table of contents now in place.

fmthard -s /mig_files/d12_0037.vtoc /dev/rdisk/c2t3d12s2

fmthard: New volume table of contents now in place.

```

The VTOC on the target devices now matches the VTOC on the source devices:

```

prtvtoc /dev/rdisk/c2t3d11s2

* /dev/rdisk/c2t3d11s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   64 sectors/track
*   15 tracks/cylinder
*   960 sectors/cylinder
*   9204 cylinders
*   9202 accessible cylinders
*
* Flags:
*   1: unmountable
*   10: read-only
*
*
* Partition  Tag  Flags      First      Sector      Last
* Partition  Tag  Flags      Sector     Count      Sector  Mount Directory
*   2         5    01         0    8833920    8833919
*   7        15    01         0    8833920    8833919

prtvtoc /dev/rdisk/c2t3d12s2

* /dev/rdisk/c2t3d12s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   64 sectors/track
*   15 tracks/cylinder
*   960 sectors/cylinder
*   9204 cylinders
*   9202 accessible cylinders
*

```

```
* Flags:
* 1: unmountable
* 10: read-only
*
*
* Partition Tag Flags First Sector Sector Count Last Sector Mount Directory
* 2 5 01 0 8833920 8833919
* 7 15 01 0 8833920 8833919
```

Use the following commands to import, start, and mount the file system to the Veritas disk group:

```
vxldg -C import geodg

vxvol -g geodg start geodgvol01

vxprint -htQq

dg geodg          default          default 29000    1203016756.36.licoa034

dm geodg01        c2t3d11s2        auto    2048     8831616  -
dm geodg02        c2t3d12s2        auto    2048     8831616  -

v geodgvol01      -                ENABLED ACTIVE    17663232 ROUND    -        fsgen
pl geodgplex01    geodgvol01    ENABLED ACTIVE    17663232 CONCAT    -        RW
sd geodg01sd01    geodgplex01    geodg01 0         8831616 0         c2t3d11 ENA
sd geodg02sd01    geodgplex01    geodg02 0         8831616 8831616 c2t3d12 ENA

fsck -F vxfs /dev/vx/dsk/geodg/geodgvol01
log replay in progress
replay complete - marking super-block as CLEAN
```

```
mount -F vxfs /dev/vx/dsk/geodg/geodgvol01 /mp/geodgvol01_R2mp
```

Use the following command to verify that the data is intact:

```
ls -l /mp/geodgvol01_R2mp
total 8388678
drwxr-xr-x 2 root other 3072 Feb 14 16:36 flat_files
drwxr-xr-x 2 root root 96 Feb 14 14:26 lost+found
-rw-----T 1 root other 1073741824 Feb 14 16:37 testfile1
-rw-----T 1 root other 1073741824 Feb 14 16:38 testfile2
-rw-----T 1 root other 1073741824 Feb 14 16:39 testfile3
-rw-----T 1 root other 1073741824 Feb 14 16:41 testfile4

ls -l /mp/geodgvol01_R2mp/flat_files
total 290
-rw-r--r-- 1 root other 131 Feb 14 16:36 arp.conf
-rw-r--r-- 1 root other 1583 Feb 14 16:36 audiocs.conf
-rw-r--r-- 1 root other 1414 Feb 14 16:36 audioens.conf
-rw-r--r-- 1 root other 1601 Feb 14 16:36 audiotcs.conf
-rw-r--r-- 1 root other 1810 Feb 14 16:36 bofi.conf
-rw-r--r-- 1 root other 135 Feb 14 16:36 clone.conf
-rw-r--r-- 1 root other 129 Feb 14 16:36 cn.conf
-rw-r--r-- 1 root other 139 Feb 14 16:36 conskbd.conf
-rw-r--r-- 1 root other 164 Feb 14 16:36 consms.conf
...

cat /mp/geodgvol01_R2mp/flat_files/arp.conf
#
# Copyright (c) 1992, by Sun Microsystems, Inc.
#
#ident "@(#)arp.conf 1.4 93/06/03 SMI"

name="arp" parent="pseudo" instance=0;
```

## Configuring Virtual Provisioning

In the following Virtual Provisioning example, a thin pool is created, DATA devices and thin devices are added, and various `symcfg list` and `show` commands are shown.

### Creating a pool

A command file (`thin_config_file`) is used with the `symconfigure` command to create the environment. The contents of `thin_config_file` are shown here:

```
create pool testpool6,
    type = thin,
    max_subs_percent= 85,
    rebalance_variance = 4,
    vp_compression = DISABLE;
```

The command file is committed with the following command:

```
symconfigure -sid 343 -f thin_config_file -v -nop commit
```

A list of pools, including the `testpool6` pool, can be viewed using the following command:

```
symcfg -sid 187 list -pools -mb
```

Symmetrix ID: 000195700187

```

              S Y M M E T R I X   P O O L S
-----
Pool      Flags  Dev      Usable      Free      Used Full Comp
Name      PTECSL Config      MBs        MBs        MBs  (%)  (%)
-----
DEFAULT_POOL SMF-EI Mixed      21094524   21094524      0      0      0
DEFAULT_POOL SM9-DI Unknown      0           0           0      0      0
DEFAULT_POOL SM8-DI Unknown      0           0           0      0      0
DEFAULT_POOL SMA-DI Unknown      0           0           0      0      0
. . .
. . . <output shortened>
raid6annel1 TFFDEI RAID-6(6+2)   1199292   1199170     122      0      0
2miranne3   TFFDEI 2-Way Mir     789462   789374      89      0      0
testpool6   T--SD- Unknown      0           0           0      0      0

Total
MBs                30128053   29828365   299688      0      0

```

Legend:

(P)ool Type:

S = Snap, R = Rdfa DSE T = Thin

(T)echnology:

S = SATA, F = Fibre Channel, E = Enterprise Flash Drive, M = Mixed, - = N/A

Dev (E)mulation:

F = FBA, A = AS400, 8 = CKD3380, 9 = CKD3390, - = N/A

(C)ompression:

E = Enabled, D = Disabled, N = Enabling, S = Disabling, - = N/A

(S)tate:

E = Enabled, D = Disabled, B = Balancing

Disk (L)ocation:

I = Internal, X = External, M = Mixed, - = N/A

I = Internal, X = External, M = Mixed, - = N/A

## Adding DATA devices

The command file, `thin_config_file2`, adds DATA devices 6ED through 710 to the pool:

```
add dev 6ED:710 to pool testpool6 type=thin
member_state=enable;
```

The file is then committed with the `symconfigure` command.

```
symconfigure -sid 187 -f thin_config_file2 -v -nop commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
{
  add dev 06ED:0710 to pool testpool6 type=thin, memberstate = Enable;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Locking devices.....Locked.
Committing configuration changes.....Reordering.
Adding pool devs .....Done.
Enabling pool devs .....Done.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

The configuration change session has successfully completed.

The next command shows the thin pool after the DATA devices are added:

```
symcfg -sid 187 -pool testpool6 -thin -detail -all -mb show
```

Symmetrix ID: 000195700187

```
Symmetrix ID           : 000195700187
Pool Name              : testpool6
Pool Type              : Thin
Disk Location          : Internal
Technology              : FC
Dev Emulation          : FBA
Dev Configuration      : RAID-6(6+2)
Pool State             : Enabled
Compression State      : Disabled
# of Devices in Pool   : 36
# of Enabled Devices in Pool : 36
# of Usable Tracks in Pool : 2484864
# of Allocated Tracks in Pool : 0
# of Tracks saved by compression : 0
# of Shared Tracks in Pool : 0
Pool Utilization (%)   : 0
Pool Compression Ratio (%) : 0
Max. Subscription Percent : 85
Rebalance Variance     : 4%
Max devs per rebalance scan : 256
Pool Reserved Capacity : None
```

Enabled Devices(36):

```
{
-----
Sym      Usable    Alloc    Free Full FLG Device
Dev      MBs      MBs      MBs  (%)  S  State
-----
06ED      4314        0    4314    0  .  Enabled
```

```

06EE      4314      0      4314      0      .      Enabled
06EF      4314      0      4314      0      .      Enabled
06F0      4314      0      4314      0      .      Enabled
06F1      4314      0      4314      0      .      Enabled
06F2      4314      0      4314      0      .      Enabled
06F3      4314      0      4314      0      .      Enabled
06F4      4314      0      4314      0      .      Enabled
06F5      4314      0      4314      0      .      Enabled
06F6      4314      0      4314      0      .      Enabled
06F7      4314      0      4314      0      .      Enabled
06F8      4314      0      4314      0      .      Enabled
06F9      4314      0      4314      0      .      Enabled
06FA      4314      0      4314      0      .      Enabled
06FB      4314      0      4314      0      .      Enabled
06FC      4314      0      4314      0      .      Enabled
06FD      4314      0      4314      0      .      Enabled
06FE      4314      0      4314      0      .      Enabled
06FF      4314      0      4314      0      .      Enabled
0700      4314      0      4314      0      .      Enabled
0701      4314      0      4314      0      .      Enabled
0702      4314      0      4314      0      .      Enabled
0703      4314      0      4314      0      .      Enabled
0704      4314      0      4314      0      .      Enabled
0705      4314      0      4314      0      .      Enabled
0706      4314      0      4314      0      .      Enabled
0707      4314      0      4314      0      .      Enabled
0708      4314      0      4314      0      .      Enabled
0709      4314      0      4314      0      .      Enabled
070A      4314      0      4314      0      .      Enabled
070B      4314      0      4314      0      .      Enabled
070C      4314      0      4314      0      .      Enabled
070D      4314      0      4314      0      .      Enabled
070E      4314      0      4314      0      .      Enabled
070F      4314      0      4314      0      .      Enabled
0710      4314      0      4314      0      .      Enabled
-----
MBs      155304      0      155304      0
}

```

No Thin Devices Bound to Device Pool testpool6

No Other-Pool Bound Thin Devices have allocations in Device Pool testpool6

Legend:

Enabled devices FLG:

(S)hared Tracks : X = Shared Tracks , . = No Shared Tracks

Bound Devices FLG:

S(T)atus : B = Bound, I = Binding, U = Unbinding, A = Allocating,  
D = Deallocating, R = Reclaiming, C = Compressing,  
N = Uncompressing, . = Unbound,

The next command looks for thin devices:

**symdev -sid 187 list -tdev -devs 1600:161F**

Symmetrix ID: 000195700187

Device Name		Directors			Device			
Sym	Physical	SA	:P	DA :IT	Config	Attribute	Sts	Cap (MB)
1600	Not Visible	???:?	NA:NA	TDEV		N/Grp'd	NR	8631
1601	Not Visible	???:?	NA:NA	TDEV		N/Grp'd	NR	8631

1602	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1603	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1604	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1605	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1606	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1607	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1608	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1609	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160A	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160B	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160C	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160D	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160E	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
160F	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1610	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1611	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1612	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1613	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1614	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1615	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1616	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1617	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1618	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
1619	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161A	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161B	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161C	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161D	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161E	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631
161F	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	8631

## Binding thin devices to the pool

To bind 32 8GB thin devices to thin pool `testpool6` and preallocate no space to 16 thin devices, and preallocate 1 GB to each of 16 of the thin device, create and commit the following command file:

```
bind tdev 1600:160F to pool testpool6;
bind tdev 1610:161F to pool testpool6 preallocate size=1GB;
```

```
symconfigure -sid 187 -f thin_config_file4 -v -nop commit
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000195700187
{
  bind tdev 1600:160F to pool testpool6 preallocate size=0 cyl ;
  bind tdev 1610:161F to pool testpool6 preallocate size=1093 cyl ;
}

Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Locking devices.....Locked.
Committing configuration changes.....Started.
Binding devices.....Done.
Allocating devices.....Started.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

## Listing thin devices in a pool

To list the thin devices use this command:

```
symcfg -sid 187 list -tdev -mb -devs 1600:161F
```

Symmetrix ID: 000195700187

Enabled Capacity (MBs) : 9037830

Bound Capacity (MBs) : 276180

S Y M M E T R I X   T H I N   D E V I C E S									
Sym	Bound Pool Name	Flgs EMPT	Total MBs	Total Allocated MBs (%)		Total Written MBs (%)		Compressed Size/Ratio MBs (%)	
1600	testpool6	F..B	8631	1	0	0	0	1	0
1601	testpool6	F..B	8631	1	0	0	0	1	0
1602	testpool6	F..B	8631	1	0	0	0	1	0
1603	testpool6	F..B	8631	1	0	0	0	1	0
1604	testpool6	F..B	8631	1	0	0	0	1	0
1605	testpool6	F..B	8631	1	0	0	0	1	0
1606	testpool6	F..B	8631	1	0	0	0	1	0
1607	testpool6	F..B	8631	1	0	0	0	1	0
1608	testpool6	F..B	8631	1	0	0	0	1	0
1609	testpool6	F..B	8631	1	0	0	0	1	0
160A	testpool6	F..B	8631	1	0	0	0	1	0
160B	testpool6	F..B	8631	1	0	0	0	1	0
160C	testpool6	F..B	8631	1	0	0	0	1	0
160D	testpool6	F..B	8631	1	0	0	0	1	0
160E	testpool6	F..B	8631	1	0	0	0	1	0
160F	testpool6	F..B	8631	1	0	0	0	1	0
1610	testpool6	F..B	8631	1026	12	0	0	1026	0
1611	testpool6	F..B	8631	1026	12	0	0	1026	0
1612	testpool6	F..B	8631	1026	12	0	0	1026	0
1613	testpool6	F..B	8631	1026	12	0	0	1026	0
1614	testpool6	F..B	8631	1026	12	0	0	1026	0
1615	testpool6	F..B	8631	1026	12	0	0	1026	0
1616	testpool6	F..B	8631	1026	12	0	0	1026	0
1617	testpool6	F..B	8631	1026	12	0	0	1026	0
1618	testpool6	F..B	8631	1026	12	0	0	1026	0
1619	testpool6	F..B	8631	1026	12	0	0	1026	0
161A	testpool6	F..B	8631	1026	12	0	0	1026	0
161B	testpool6	F..B	8631	1026	12	0	0	1026	0
161C	testpool6	F..B	8631	1026	12	0	0	1026	0
161D	testpool6	F..B	8631	1026	12	0	0	1026	0
161E	testpool6	F..B	8631	1026	12	0	0	1026	0
161F	testpool6	F..B	8631	1026	12	0	0	1026	0
Total			-----	-----	---	-----	---	-----	---
MBs			276180	16428	0	0	0	16428	0

### Legend:

Flags: (E)mulation : A = AS400, F = FBA, 8 = CKD3380, 9 = CKD3390  
 (M)ultipool : X = multi-pool allocations, . = single pool allocation  
 (P)ersistent Allocs : A = All, S = Some, . = None  
 S(T)atus : B = Bound, I = Binding, U = Unbinding, A = Allocating,  
 D = Deallocating, R = Reclaiming, C = Compressing,  
 N = Uncompressing, . = Unbound,



## Showing all the pool details

Use the `symcfg show pool -thin` command with the `-all` and `-detail` options to see the pool configuration as shown in the following command and output:

```
symcfg -sid 187 -pool testpool6 -thin -all -mb -detail show
```

```
Symmetrix ID: 000195700187
```

```
Symmetrix ID           : 000195700187
Pool Name              : testpool6
Pool Type              : Thin
Disk Location          : Internal
Technology             : FC
Dev Emulation          : FBA
Dev Configuration     : RAID-6(6+2)
Pool State             : Enabled
Compression State      : Disabled
# of Devices in Pool   : 36
# of Enabled Devices in Pool : 36
# of Usable Tracks in Pool : 2484864
# of Allocated Tracks in Pool : 262848
# of Tracks saved by compression : 0
# of Shared Tracks in Pool : 0
Pool Utilization (%)   : 10
Pool Compression Ratio (%) : 0
Max. Subscription Percent : 250
Rebalance Variance     : 4%
Max devs per rebalance scan : 256
Pool Reserved Capacity : None
```

```
Enabled Devices(36):
```

```
{
```

Sym Dev	Usable MBs	Alloc MBs	Free MBs	Full (%)	FLG S	Device State
06ED	4314	457	3857	10	.	Enabled
06EE	4314	457	3857	10	.	Enabled
06EF	4314	458	3857	10	.	Enabled
06F0	4314	457	3857	10	.	Enabled
06F1	4314	456	3858	10	.	Enabled
06F2	4314	457	3857	10	.	Enabled
06F3	4314	456	3858	10	.	Enabled
06F4	4314	456	3858	10	.	Enabled
06F5	4314	457	3857	10	.	Enabled
06F6	4314	455	3859	10	.	Enabled
06F7	4314	456	3858	10	.	Enabled
06F8	4314	455	3859	10	.	Enabled
06F9	4314	456	3858	10	.	Enabled
06FA	4314	455	3859	10	.	Enabled
06FB	4314	455	3859	10	.	Enabled
06FC	4314	455	3859	10	.	Enabled
06FD	4314	456	3858	10	.	Enabled
06FE	4314	456	3858	10	.	Enabled
06FF	4314	456	3858	10	.	Enabled
0700	4314	457	3857	10	.	Enabled
0701	4314	458	3857	10	.	Enabled
0702	4314	457	3857	10	.	Enabled
0703	4314	457	3857	10	.	Enabled
0704	4314	457	3857	10	.	Enabled
0705	4314	456	3858	10	.	Enabled
0706	4314	457	3857	10	.	Enabled
0707	4314	456	3858	10	.	Enabled
0708	4314	457	3857	10	.	Enabled

```

0709      4314      456      3858    10 . Enabled
070A      4314      456      3858    10 . Enabled
070B      4314      457      3857    10 . Enabled
070C      4314      457      3857    10 . Enabled
070D      4314      457      3857    10 . Enabled
070E      4314      457      3857    10 . Enabled
070F      4314      457      3857    10 . Enabled
0710      4314      457      3857    10 . Enabled
-----
MBs        155304      16428      138876    10
}

```

Pool Bound Thin Devices(32):

```

{
-----
Sym   FLG      Pool      Pool      Total      Compressed
Dev   T      Total Sub   Allocated  Written   Size/Ratio
      MBs (%)   MBs (%)   MBs (%)   MBs (%)   MBs (%)
-----
1600   B      8631   6      1   0      0   0      1   0
1601   B      8631   6      1   0      0   0      1   0
1602   B      8631   6      1   0      0   0      1   0
1603   B      8631   6      1   0      0   0      1   0
1604   B      8631   6      1   0      0   0      1   0
1605   B      8631   6      1   0      0   0      1   0
1606   B      8631   6      1   0      0   0      1   0
1607   B      8631   6      1   0      0   0      1   0
1608   B      8631   6      1   0      0   0      1   0
1609   B      8631   6      1   0      0   0      1   0
160A   B      8631   6      1   0      0   0      1   0
160B   B      8631   6      1   0      0   0      1   0
160C   B      8631   6      1   0      0   0      1   0
160D   B      8631   6      1   0      0   0      1   0
160E   B      8631   6      1   0      0   0      1   0
160F   B      8631   6      1   0      0   0      1   0
1610   B      8631   6     1026  12      0   0     1026   0
1611   B      8631   6     1026  12      0   0     1026   0
1612   B      8631   6     1026  12      0   0     1026   0
1613   B      8631   6     1026  12      0   0     1026   0
1614   B      8631   6     1026  12      0   0     1026   0
1615   B      8631   6     1026  12      0   0     1026   0
1616   B      8631   6     1026  12      0   0     1026   0
1617   B      8631   6     1026  12      0   0     1026   0
1618   B      8631   6     1026  12      0   0     1026   0
1619   B      8631   6     1026  12      0   0     1026   0
161A   B      8631   6     1026  12      0   0     1026   0
161B   B      8631   6     1026  12      0   0     1026   0
161C   B      8631   6     1026  12      0   0     1026   0
161D   B      8631   6     1026  12      0   0     1026   0
161E   B      8631   6     1026  12      0   0     1026   0
161F   B      8631   6     1026  12      0   0     1026   0
-----
MBs        276180  178      16428   6          0   0      16428   0
}

```

No Other-Pool Bound Thin Devices have allocations in Device Pool testpool6

Legend:

Enabled devices FLG:

(S)hared Tracks : X = Shared Tracks , . = No Shared Tracks

Bound Devices FLG:

S(T)atus : B = Bound, I = Binding, U = Unbinding, A = Allocating,  
D = Deallocating, R = Reclaiming, C = Compressing,  
N = Uncompressing, . = Unbound,

## Automatically creating metadevices

The following output shows the Auto Meta feature is enabled on Symmetrix array 069. If a new volume of 120000 cylinders or larger is created, the Auto Meta feature is invoked, creating a striped meta volume with each member being 65520 cylinders in size. The total metavolume size is rounded up based on the member size.

```
symcfg -sid 69 list -v

Symmetrix ID: 000195600069
Time Zone   : EST

Product Model      : VMAX
Symmetrix ID       : 000195600069

.....

Auto Meta          : Enabled
Minimum Auto Meta Size : 120000
Auto Meta Member Size  : 65520
Auto Meta Configuration : Striped
```

In the following example, a size of 160000 cylinders is requested. This rounds up the size to a total of 196560 cylinders. Thus creating three volumes each at 65520 cylinders, which are combined to form the metavolume.

```
symconfigure -sid 69 -cmd "create dev count=1, size=160000 cyl, emulation=FBA,
config=2-Way-Mir, disk_group=1;" commit
```

```
Execute a symconfigure operation for symmetrix '000195600069' (y/[n]) ? y
```

```
A Configuration Change operation is in progress. Please wait...
```

```
Establishing a configuration change session.....Established.
Processing symmetrix 000195600069
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 013 of 063 steps.....Executing.
Step 036 of 167 steps.....Executing.
Step 057 of 167 steps.....Executing.
Step 057 of 167 steps.....Executing.
Step 118 of 174 steps.....Executing.
Step 121 of 174 steps.....Executing.
Step 121 of 174 steps.....Executing.
Step 145 of 174 steps.....Executing.
Step 161 of 174 steps.....Executing.
Local: COMMIT.....Done.

New symdevs: 0988:098A [Striped meta, head 0988, member_size 65520 cyl]
Terminating the configuration change session.....Done.
```

```
The configuration change session has successfully completed.
```

The following output shows the newly-created striped metavolume:

```
symdev -sid 69 show 988

Device Physical Name      : Not Visible
Device Symmetrix Name     : 0988
```

```

Device Serial ID      : N/A
Symmetrix ID         : 000195600069

Number of RAID Groups : 1
Encapsulated Device   : No

Attached BCV Device   : N/A

Attached VDEV TGT Device : N/A

Vendor ID             : EMC
Product ID            : SYMMETRIX
Product Revision      : 5876
Device WWN            : 60000970000195600069533030393838
Device Emulation Type : FBA
Device Defined Label Type: N/A
Device Defined Label   : N/A
Device Sub System Id  : 0x0001
Cache Partition Name  : DEFAULT_PARTITION

Device Block Size     : 512

Device Capacity
{
  Cylinders      : 196560
  Tracks         : 2948400
  512-byte Blocks : 377395200
  MegaBytes      : 184275
  KiloBytes      : 188697600

  Geometry Limited : No
}

Device External Identity
{
  Device WWN      : 60000970000195600069533030393838

  Front Director Paths (0): N/A

  Geometry          : Native
  {
    Sectors/Track      : 128
    Tracks/Cylinder    : 15
    Cylinders          : 196560
    512-byte Blocks    : 377395200
    MegaBytes          : 184275
    KiloBytes          : 188697600
  }
}

Device Configuration : 2-Way Mir      (Meta Head)

Device is WORM Enabled : No
Device is WORM Protected : No

SCSI-3 Persistent Reserve: Enabled

Dynamic Spare Invoked : No

Dynamic RDF Capability : None

STAR Mode              : No
STAR Recovery Capability : None
STAR Recovery State    : NA

Device Service State   : Normal

```

```

Device Status           : Ready           (RW)
Device SA Status        : N/A             (N/A)
Device User Pinned      : False
Host Access Mode        : Active
Device Tag(s)           : None

```

```

Extent Based Clone      : None
DIF1 Flag               : False
AS400_GK                : False
Host Cache Registered   : False

```

```

Meta Configuration      : Striped
Meta Stripe Size        : 960k           (1 Cylinders)
Meta Device Members (3) :

```

```

{
-----
                        BCV  DATA                        RDF  DATA
-----
Sym   Cap   Std Inv BCV Inv Pair      R1 Inv R2 Inv Pair
Dev   (MB)   Tracks Tracks State      Tracks Tracks State
-----
--> 0988 61425      -   -  N/A          -   -  N/A
    0989 61425      -   -  N/A          -   -  N/A
    098A 61425      -   -  N/A          -   -  N/A
-----
          184275      -   -          -   -
}

```

```

Mirror Set Type          : [RAID-1,N/A,N/A,N/A]

```

```

Mirror Set DA Status     : [RW,N/A,N/A,N/A]

```

```

Mirror Set Inv. Tracks   : [0,0,0,0]

```

#### Back End Disk Director Information

```

{
Hyper Type               : RAID-1
Hyper Status             : Ready           (RW)
Disk [Director, Interface, TID] : [N/A,N/A,N/A]
Spindle ID               : N/A
Disk Director Volume Number : N/A
Hyper Number             : N/A
Mirror Number            : 1
Disk Group Number        : 1
Disk Group Name          : DISK_GROUP_001
}

```

#### RAID Group Information

```

{
Mirror Number            : 1
RAID Type                : RAID-1
Device Position          : Primary
Protection Level         : 2
RAID Group Service State : Normal
Failing Member Mask      : N/A
Hyper Devices:

```

```

{
Device : 0988 (M)

```

```

{
-----
Spindle  Disk   DA      Hyper      Member      Disk
          DA :IT  Vol#   Num Cap(MB) Num Status Grp#  Cap(MB)
-----
19C9     08D:C2  102   53   62388   1  RW      1  1823565
A8D      07B:D2  357   53   62388   2  RW      1  1823565

```

```

    }
    Device : 0989 (m)
    {
    -----
    Spindle   Disk       DA       Hyper      Member      Disk
              DA :IT     Vol#     Num Cap(MB) Num Status Grp#  Cap(MB)
    -----
    385        08A:D2      359     52    62388    1 RW      1  1823565
    11D1       07C:C2      101     52    62388    2 RW      1  1823565
    }
    Device : 098A (m)
    {
    -----
    Spindle   Disk       DA       Hyper      Member      Disk
              DA :IT     Vol#     Num Cap(MB) Num Status Grp#  Cap(MB)
    -----
    198F       07D:D6      445     50    62388    1 RW      1  1823565
    ACB        08B:C6      194     50    62388    2 RW      1  1823565
    }
  }
}

```

## Setting port flags on a front-end director

The following command sets the port flags on a front-end director of Symmetrix array 700181:

```
symconfigure -sid 181 -cmd " set port 6E:0 Disable_Q_Reset_on_UA=enable; " commit
```

```
Execute a symconfigure operation for symmetrix '000195700181' (y/[n]) ? y
```

A Configuration Change operation is in progress. Please wait...

```

Establishing a configuration change session.....Established.
Processing symmetrix 000195700181
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 008 of 063 steps.....Executing.
Step 013 of 063 steps.....Executing.
Step 032 of 063 steps.....Executing.
Step 036 of 063 steps.....Executing.
Step 048 of 101 steps.....Executing.
Step 068 of 101 steps.....Executing.
Step 071 of 101 steps.....Executing.
Step 077 of 101 steps.....Executing.
Step 081 of 101 steps.....Executing.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.

```

The configuration change session has successfully completed.

## Removing port flags from a front-end director

The following command removes the port flags (set in the previous example) from a front-end director of Symmetrix array 700181:

```
symconfigure -sid 181 -cmd " set port 6E:0 Avoid_Reset_Broadcast=disable; " commit
```

```
Execute a symconfigure operation for symmetrix '000195700181' (y/[n]) ? y
```

A Configuration Change operation is in progress. Please wait...

```
Establishing a configuration change session.....Established.
Processing symmetrix 000195700181
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Initiating COMMIT of configuration changes.....Queued.
COMMIT requesting required resources.....Obtained.
Step 008 of 063 steps.....Executing.
Step 013 of 063 steps.....Executing.
Step 035 of 063 steps.....Executing.
Step 036 of 063 steps.....Executing.
Step 040 of 101 steps.....Executing.
Step 057 of 101 steps.....Executing.
Step 069 of 101 steps.....Executing.
Step 075 of 101 steps.....Executing.
Step 078 of 101 steps.....Executing.
Step 099 of 101 steps.....Executing.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.





# APPENDIX B

## Auto-provisioning Groups Example

This appendix describes how to create and display the elements that can comprise a masking view. This appendix contains the following topics:

- ◆ [Creating and displaying children and parent storage groups .....](#) 434
- ◆ [Creating and displaying port groups .....](#) 435
- ◆ [Creating and showing initiator groups .....](#) 436
- ◆ [Creating and displaying masking views .....](#) 437

## Creating and displaying children and parent storage groups

The following command creates a storage group named `Child_SG1` that contains volumes 200 through 204:

```
symaccess -sid 69 -type storage -name Child_SG1 create dev 200:204
```

The following command creates a storage group named `Child_SG2` that contains volumes 300 to 304:

```
symaccess -sid 69 -type storage -name Child_SG2 create dev 300:304
```

The following command creates a storage group named `Parent_SG`, which becomes the parent in a cascaded storage group containing the children storage groups `Child_SG1` and `Child_SG2`. `Child_SG1` and `Child_SG2` become children to `Parent_SG` from this command:

```
symaccess -sid 69 -type storage -name Parent_SG create sg  
Child_SG1,Child_SG2
```

## Showing parent storage groups

The following output shows that `Child_SG1` and `Child_SG2` are children of `Parent_SG`:

```
symaccess -sid 69 -type storage show Parent_SG

Symmetrix ID           : 000195600069

Storage Group Name      : Parent_SG
Last updated at         : 01:12:29 PM on Mon Feb 13, 2012

Number of Storage Groups : 2
Storage Group Names      : Child_SG1                (IsChild)
                        : Child_SG2                (IsChild)

Devices                 : 0200:0204 0300:0304

Masking View Names
{
  None
}
```

## Showing children storage groups

The following output shows that Parent\_SG is a parent of Child\_SG1:

```
symaccess -sid 69 -type storage show Child_SG1

Symmetrix ID           : 000195600069

Storage Group Name      : Child_SG1
Last updated at        : 01:10:17 PM on Mon Feb 13,2012

Number of Storage Groups : 1
Storage Group Names     : Parent_SG                      (IsParent)

Devices                : 0200:0204

Masking View Names
{
  None
```

## Creating and displaying port groups

The following command creates a port group named `My_portgroup` and adds director port 7E:0 to it:

```
symaccess -sid 69 create -type port -name My_portgroup -dirport 7E:0
```

## Displaying port group information

The following output shows that director port 7E:0 was successfully added to `My_portgroup`:

```
symaccess -sid 69 -type port show My_portgroup

Symmetrix ID           : 000195600069

Port Group Name        : My_portgroup
Last updated at        : 01:23:38 PM on Mon Feb 13,2012

Director Identification
{
  FA-7E:0
}

Masking View Names
{
  None
}
```

The following output shows the Port Count and View Count of `My_portgroup`:

```
symaccess -sid 69 -type port -name My_portgroup list

Symmetrix ID           : 000195600069
```

Port Group Name	Port Count	View Count
My_portgroup	1	0

## Creating and showing initiator groups

The following command creates an initiator group named `My_server` and adds an initiator with a WWN of `10000000c990e075` to it:

```
symaccess -sid 69 create -type init -name My_server -wwn  
10000000c990e075
```

## Displaying initiator group information

The following output shows that the host initiator with the WWN of `10000000c990e075` was successfully added to `My_server`:

```
symaccess -sid 69 -type init show My_server

Symmetrix ID           : 000195600069

Initiator Group Name   : My_server
Last updated at        : 01:29:53 PM on Mon Feb 13,2012

Host Initiators
{
  WWN    :10000000c990e075 [alias: 14se0066/10000000c990e075]
}

Masking View Names
{
  None
}

Parent Initiator Groups
{
  None
}
```

The following output provides more details about `My_server`:

```
symaccess -sid 69 -type init show My_server -det

Symmetrix ID           : 000195600069

Initiator Group Name   : My_server
Last updated at        : 01:29:53 PM on Mon Feb 13,2012

Port Flag Overrides    : No
Consistent Lun         : No

Originator Port wwn    : 10000000c990e075
User-generated Name    : 14se0066/10000000c990e075
FCID Lockdown          : No
Heterogeneous Host     : No
Port Flag Overrides    : No
CHAP Enabled           : N/A
Type                   : Fibre
```

The following output shows the Init Count and View Count of My\_server:

```
symaccess -sid 69 -type init -name My_server list
```

Symmetrix ID : 000195600069

Initiator Group Name	Init Count	View Count
My_server	1	0

## Creating and displaying masking views

The following command creates a masking view named My\_View from the previously created storage, initiator and port groups.

```
symaccess -sid 69 create view -name My_View -ig My_server -sg Parent_SG -pg My_portgroup
```

## Showing masking view information

The following output shows that My\_View contains the Host Initiator with the WWN of 10000000c990e075, the Port Group My\_portgroup, and the Storage Group Parent\_SG with children Child\_SG1 and Child\_SG2:

```
symaccess -sid 69 view show My_View
```

Symmetrix ID : 000195600069

Masking View Name : My\_View  
Last updated at : 01:35:11 PM on Mon Feb 13,2012

Initiator Group Name : My\_server

**Host Initiators**

```
{
  WWN : 10000000c990e075 [alias: 14se0066/10000000c990e075]
}
```

**Port Group Name : My\_portgroup**

Director Identification

```
{
  FA-7E:0
}
```

**Storage Group Name : Parent\_SG**

Number of Storage Groups : 2  
Storage Group Names : Child\_SG1 (IsChild)  
Child\_SG2 (IsChild)

Sym Dev	Dir:P	Physical Device Name	Host Lun	Attr	Cap(MB)
0200	07E:0	Not Visible	1		3
0201	07E:0	Not Visible	2		3
0202	07E:0	Not Visible	3		3
0203	07E:0	Not Visible	4		3
0204	07E:0	Not Visible	5		3
0300	07E:0	Not Visible	6		246251
0301	07E:0	Not Visible	7		246251

## Auto-provisioning Groups Example

0302	07E:0	Not Visible	8	246251
0303	07E:0	Not Visible	9	246251
0304	07E:0	Not Visible	a	246251
				-----
Total Capacity				1231270

# APPENDIX C

## FAST Output Examples

This chapter provides several examples of FAST output and explains the column headings.

- ◆ Showing associations for mixed storage groups using -all ..... 440
- ◆ Showing associations for standard tiers ..... 441
- ◆ Listing the FAST plan for disk groups with the -v option ..... 442
- ◆ Listing the history with the -v option..... 443
- ◆ Listing the technology demand for DP tiers..... 444
- ◆ Listing the technology demand for VP tiers using -v ..... 446
- ◆ Listing the technology demand for DP tiers using -v ..... 449
- ◆ Listing the technology demand using V71 mode ..... 451

## Showing associations for mixed storage groups using -all

The show association display includes a new `Tier Type` column in the tiers table. When the `-all` option is specified, all standard and thin devices in the storage group are shown, including those not managed by FAST (by default, only the devices managed by FAST would display). The `Policy Tier Name` displays N/A for such devices.

To show the associated policies and tiers for storage group `sg_fast` on Symmetrix 601, enter:

```
symfast -sid 601 show -association -sg sg_fast -all
```

Output similar to the following displays:

```
Symmetrix ID           : 000192600601
```

```
Storage Group          : sg_fast
```

```
Thin Devices(1)
```

```
{
-----
      Flgs Dev      Total Bound      Allocated
Sym   PC  Emul      Tracks Pool Name      Tracks
-----
00BA  N.  FBA        33000 test1        33012

Total
Tracks      -----
      33000      33012
      GBs        2        2
}
```

```
Standard Devices(1)
```

```
{
-----
      Flags      Policy      Dsk  Dsk      Cap
Sym   P  Protection  Tier Name  Grp  Group Name  Tech  (GB)
-----
01A9  N  Unprotected  N/A        004  DISK_GROUP_004  FC      2

Total
      GBs      -----
      2
}
```

```
Policy Name           : test_fp
Priority               : 2
RDF Coordination      : Disabled
```

```
Tiers(1)
```

```
{
-----
                                L
                                Max SG O
Tier Name                    Type  Percent C Tech  Target      Flgs
-----
Test_tier                    VP      100 I FC    RAID-1      .
}
```

Legend:

Tier Type: DP = Disk Group Provisioning, VP = Virtual Pools

Device Flags:

(P)inned : Y = Device is Pinned, N = Device is not Pinned

(C)ompression : X = Device is Compressed, . = Device is not Compressed

Tier Flags:

Disk (Loc)ation : I = Internal, X = External



(C)ompression : X = Compression Capable, . = Not Compression Capable

## Showing associations for standard tiers

To show the association for storage group `test` on Symmetrix 601, enter:

```
symfast -sid 01 show -association -sg test
```

Output similar to the following displays:

Symmetrix ID : 000192600601

Storage Group : test

Standard Devices(1)

```
{
-----
      Flags
Sym   P   Protection   Policy   Dsk   Dsk   Tech   Cap
      P   Protection   Tier Name   Grp   Group Name   (GB)
-----
021B   N   RAID-1       test_dp     002   DISK_GROUP_002   FC       0

Total
GBs                               -----
                                0
}
```

Policy Name : test\_fp1

Priority : 2

RDF Coordination : N/A

Tiers(1)

```
{
-----
                                L
                                Max SG O
Tier Name                        Type   Percent C Tech   Target   Flgs
                                Type   Percent C Tech   Protection   C
-----
tier_dp                          DP      100 I FC    RAID-1     .
}
```

Legend:

Tier Type: DP = Disk Group Provisioning, VP = Virtual Pools

Device Flags:

(P)inned : Y = Device is Pinned, N = Device is not Pinned

(C)ompression : X = Device is Compressed, . = Device is not Compressed

Tier Flags:

Disk (Loc)ation : I = Internal, X = External

(C)ompression : X = Compression Capable, . = Not Compression Capable

## Listing the FAST plan for disk groups with the -v option

This example show the command and output for a FAST plan (for disk group provisioning) that was automatically generated:

```
symfast -sid 234 list -plan -v
```

```
Symmetrix ID           : 000194900234

Plan ID                : 12222009:154359
Plan Type              : Auto Generated
Plan State             : CnfigInProgress
Start Time             : Tue Dec 22 20:30:23 2009
Percent Complete       : 5%
Estimated time to completion : 04:12:30
Number of Groups       : 3

. . .

Group 3:
{
  Group Attributes      : FAST Generated(Compliance)
  Group State           : NotStarted
  Time Started          : N/A
  Time Completed        : N/A
  Percent Complete      : 0%
  Estimated time to completion : 04:12:30

  Move Devices(6)
  {
    SRC Device(s)       : 0042 0043 0044
    SRC Protection Type  : R6(14+2)
    SRC Storage Group Name : OraSales
    SRC Tier Name        : N/A
    SRC Disk Group Number : 3
    SRC Disk Group Name   : fiber_disks
    TGT Protection Type  : R5(7+1)
    TGT Tier Name        : ArchiveDBTier
    TGT Disk Group Number : 4
    TGT Disk Group Name   : sata_disks
    TGT Disks            : [16A, D, 1] [01A, C, E] [15C, C, D] [15D, D, A]
                        : [01C, D, 2] [12A, D, D] [15B, C, E] [07A, C, 2]
                        : [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]
                        : [01F, C, 5] [12H, C, 1] [07H, D, 1] [07F, D, 8]

    Percent Complete    : 0%

    SRC Device(s)       : 0059 005A 005B
    SRC Protection Type  : R5(3+1)
    SRC Storage Group Name : OraSales
    SRC Tier Name        : PrimeDBTier
    SRC Disk Group Number : 2
    SRC Disk Group Name   : flash_disks2
    TGT Protection Type  : R5(3+1)
    TGT Tier Name        : WorkDBTier
    TGT Disk Group Number : 3
    TGT Disk Group Name   : fiber_disks
    TGT Disks            : [16C, D, 1] [01B, C, E] [15A, C, D] [15C, D, A]
                        : [01A, D, 2] [12B, D, D] [15C, C, E] [07B, C, 2]
                        : [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]

    Percent Complete    : 0%
  }
}
```

## Listing the history with the -v option

The following sample command and output lists the activity history of Symmetrix 234 using the -v (verbose) option:

```
symfast list -sid 234 -history -v
```

```
Symmetrix ID          : 000194900234
Number of Groups      : 4

Group 1:
{
  Time Started        : Tue Dec 22 09:30:23 2009
  Time Completed      : Tue Dec 22 09:35:24 2009
  Group Attributes    : Optimizer Manual Swap

  Swap Pairs (2)
  {
    SRC Device         : 0020
    SRC Disk Group Num : 4
    SRC Disk Group Name : sata_disks
    SRC protection Type : R5(3+1)
    SRC Disks          : [01A, C, E] [16B, D, D] [01C, D, 9] [16D, C, 4]
    TGT Device         : 0086
    TGT Disk Group Num : 2
    TGT Disk Group Name : flash_disks
    TGT protection Type : R6(6+2)
    TGT Disks          : [16A, D, 3] [01B, C, 1] [16B, D, F] [01C, C, 6]
                      [15A, D, 3] [02B, C, 1] [16B, C, 6] [01C, C, 5]

    . . .

Group 4:
{
  Time Started        : Tue Dec 22 20:50:23 2009
  Time Completed      : Tue Dec 22 21:05:20 2009
  Attributes          : FAST Generated(Compliance)

  Move Devices(6)
  {
    SRC Device(s)      : 0042
    SRC Protection Type : R6(14+2)
    SRC Storage Group Name : OraSales
    SRC Tier Name       : N/A
    SRC Disk Group Number : 3
    SRC Disk Group Name  : fiber_disks
    TGT Protection Type  : R5(7+1)
    TGT Tier Name       : ArchiveDBTier
    TGT Disk Group Number : 4
    TGT Disk Group Name  : sata_disks
    TGT Disks           : [16A, D, 1] [01A, C, E] [15C, C, D] [15D, D, A]
                      [01C, D, 2] [12A, D, D] [15B, C, E] [07A, C, 2]

    SRC Device(s)      : 0043
    SRC Protection Type : R6(14+2)
    SRC Storage Group Name : OraSales
    SRC Tier Name       : N/A
    SRC Disk Group Number : 3
    SRC Disk Group Name  : fiber_disks
    TGT Protection Type  : R5(7+1)
    TGT Tier Name       : ArchiveDBTier
    TGT Disk Group Number : 4
    TGT Disk Group Name  : sata_disks
    TGT Disks           : [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]
                      [01F, C, 5] [12H, C, 1] [07H, D, 1] [07F, D, 8]
```

```

. . .
SRC Device(s)           : 005B
SRC Protection Type     : R5(3+1)
SRC Storage Group Name  : OraSales
SRC Tier Name           : PrimeDBTier
SRC Disk Group Number   : 2
SRC Disk Group Name     : flash_disks2
TGT Protection Type     : R5(3+1)
TGT Tier Name           : WorkDBTier
TGT Disk Group Number   : 3
TGT Disk Group Name     : fiber_disks
TGT Disks               : [01D, C, 5] [12C, C, 1] [15B, D, 1] [07A, D, 8]
}
}

```

## Listing the technology demand for DP tiers

The next example shows a the command and ouput for DP tiers:

```
symfast list -sid 234 -tech FC -demand -dp
```

```
Symmetrix ID           : 000194900234
```

```
Technology              : FC
Raw Total(GB)           :          6000
Raw Unconfigured (GB)   :          1030
Raw Configured (GB)     :          4970
Raw FAST SG Usage Total (GB) :          4239
Raw FAST Available (GB) :          5270
Raw Max SG Demand Total (GB) :          4120
Raw Excess (GB)         :          +1150
```

```
DP Tiers (2)
```

```
{
-----
              A
              T
              T Target   Raw Tier ----- Logical Capacities (GB) -----
Tier          R Prot    (GB)   Config   Usage   Avail   Max SG   Excess
-----
WorkDBTier    F R6(6+2)   1030   1598   1420   2193   1440   +753
WorkTier      F R5(3+1)   1030   1980   1760   2533   1650   +833
[OutOfTier]   - N/A       -      150    0      0      -      -
Total                                     -----
                                         3728   3180               3090
}
```

Legend:

```
ATTR           : F = Tier in a FAST policy associated with SG(s)
                : P = Tier in a FAST policy unassociated with SG(s)
                : N = Tier not in any FAST policy
```

Definitions of each item in the report follows:

### Technology block definitions

**Raw Total** — Sum of capacities of all disks that match the technology type. Also the sum of **Raw Unconfigured** and **Raw Configured**.

**Raw Unconfigured** — Sum of unconfigured space on all disks that match the technology type.

**Raw Configured** — Sum of configured space on all disks that match the technology type.

**Raw FAST SG Usage Total** — Sum of hyper sizes of all standard devices that reside on this technology and are in a storage group associated with a FAST policy containing DP tiers.

**Raw FAST Available** — Based on **Raw FAST SG Usage Total**. If the `swap_notvisible_devs` control parameter is **ENABLED**, the hyper sizes of not-host-visible (unmapped/unmasked) standard devices that reside on this technology are included in **Raw FAST Available**. If the `allow_only_swap` control parameter is **DISABLED**, **Raw Unconfigured** is included in **Raw FAST Available**.

**Raw Max SG Demand Total** — Sum of hyper sizes of all standard devices in a storage group associated with a FAST policy containing DP tiers if they were to occupy the full allotted quota (as per the FAST policy) of space in a DP tier of this technology type. This is an estimated number. Solutions Enabler approximates the physical space that would be occupied by the standard devices based on the tier protection type.

**Raw Excess** — Difference between **Raw FAST Available** and **Raw Max FAST SG Demand**.

## Tier block definitions

**Name** — Shows names of all DP tiers of this technology type.

**ATTR** — Shows the status of the DP tier. Tiers can have 1 of 3 possible attributes:

- In a FAST policy associated with a storage group
- In a FAST policy or policies where none of the FAST policies are associated with a storage group
- Not in any FAST policy

**Target Prot** — Target protection of the DP tier.

**Raw Tier Unconfigured** — Unconfigured space in the tier disk groups. If DP tiers overlap, the unconfigured space from disk groups included in both tiers will be attributed to each tier, therefore the same free space may be counted multiple times. If a disk group does not have enough usable disks to support devices of the tier protection type (disk count of 7 for a RAID-5 (7+1) Tier), the **Raw Free** capacity for that disk group will be reported as 0.

**Logical Tier Configured** — Sum of the logical capacity of all devices that match the Tier protection type and reside on the tier disk groups. Matches the **Logical Config** column from the `symtier list -dp display`.

**Logical FAST SG Usage** — Sum of the logical capacity of all devices that match the tier protection type, reside on the tier disk groups, and are in a storage group associated with a FAST policy containing DP tiers (a subset of **Logical Tier Configured**).

**Logical FAST Available** — Based on **Logical FAST SG Usage** for the DP tier. If the tier is in a FAST policy associated with a storage group and the `swap_notvisible_devs` control parameter is **ENABLED**, the logical capacity of not-host-visible (unmapped/unmasked) standard devices that reside on this technology are included in **Logical FAST Available**. If the tier is in a FAST policy

associated with a storage group and the `allow_only_swap` control parameter is `DISABLED`, the estimated additional available capacity will be included, based on `Raw Tier Unconfigured capacity`. `Raw Tier Unconfigured capacity` is multiplied by a factor based on tier protection type to estimate how many more logical GB worth of devices FAST could move to the tier. This value is only an estimate, and it is not guaranteed that the full `FAST Available Capacity` can be reached for the tier, because the estimate does not account for:

- Affinity groups.
- The physical layout of hypervisors on the disks.
- If DP tiers overlap, the unconfigured space from disk groups included in both tiers will be attributed to each tier, therefore the same unconfigured space may be counted multiple times.

**Logical Max SG Demand** — Sum of logical capacities of all standard devices in a storage group associated with a FAST policy containing this DP tier if the devices were to occupy the full allotted quota (as per the FAST policy) of space in the DP tier. The **Logical Max SG Demand** also equals the sum of the values in the **Max SG Demand** column for all entries for this tier in the compliance report. If the tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then this value is not applicable.

**Logical Excess** — Difference between **Logical FAST Available** and **Logical Max FAST SG Demand**. If the DP tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then this value is not applicable.

---

**Note: 1** If on a given technology there exist standard devices that do not belong to any DP tier, they will be shown as `[OutOfTier]`. This can happen when the RAID type of standard devices does not match any tier RAID type, or when the defined DP tiers only contain a subset of the disk groups of the technology.

---



---

**Note: 2** Totals are not shown for the `Raw Tier Unconfig` and `FAST Avail` columns, because unconfigured space in disk groups shared by multiple tiers will be counted towards each tier, therefore it would be counted multiple times in the total.

---

## Listing the technology demand for VP tiers using -v

The following example shows a technology report for VP tiers with the `-v` (verbose) output. An explanation of the report follows the sample output:

```
symfast list -sid 234 -tech FC -demand -vp -v

Symmetrix ID                : 000194900234

Technology                   : FC
Logical Tier Enabled Total (GB) :          700
Logical Tier Free Total (GB)   :          470
Logical Tier Used Total (GB)   :          230
Logical FAST SG Usage Total (GB) :          181
Logical FAST Available Total (GB) :          495
Logical Max SG Demand Total (GB) :          235
Logical Excess (GB)           :          +260
```

```

VP Tiers (3)
{
Tier Name                : FC_R6_VPTier
Target Prot              : R6(6+2)
Logical Tier Enabled (GB) :      110
Logical Tier Free (GB)   :       90
Logical PRC Total (GB)   :       11
Logical Tier Used (GB)    :       20
Logical FAST SG Usage Total (GB) :    20
Logical FAST Available (GB) :     99
Logical Max SG Demand Total (GB) :    60
Logical Excess (GB)      :     +39
Tier Status : Tier in a FAST policy associated with SG(s)

```

## Storage Groups (1)

```

{
-----
          P FAST SG   Max SG
          r   Usage   Demand
SG Name    FAST    i Log (GB) Log (GB)
-----
SQLServer  DB_VP_Policy 1      20      60
          Total                20      60
}

```

```

Tier Name                : Finance_VPTier
Target Prot              : R5(3+1)
Logical Tier Enabled (GB) :     440
Logical Tier Free (GB)   :     280
Logical PRC Total (GB)   :      44
Logical Tier Used (GB)    :     160
Logical FAST SG Usage Total (GB) :    160
Logical FAST Available (GB) :    396
Logical Max SG Demand Total (GB) :    200
Logical Excess (GB)      :    +196
Tier Status : Tier in a FAST policy associated with SG(s)

```

## Storage Groups (1)

```

{
-----
          P FAST SG   Max SG
          r   Usage   Demand
SG Name    FAST    i Log (GB) Log (GB)
-----
Finance2010 Finance_VP_P* 1    160    200
          Total                160    200
}

```

```

Tier Name                : TierNotInPolicy
Target Prot              : R5(3+1)
Logical Tier Enabled (GB) :     150
Logical Tier Free (GB)   :     100
Logical PRC Total (GB)   :      15
Logical Tier Used (GB)    :      50
Logical FAST SG Usage Total (GB) :      0
Logical FAST Available (GB) :      0
Logical Max SG Demand Total (GB) :      -
Logical Excess (GB)      :      -
Tier Status : Tier not in any FAST policy

```

No devices in a FAST SG on tier TierNotInPolicy

```

Tier Name                : [OutOfTier]

```

```

Target Prot                               : N/A
Logical Tier Enabled (GB)                 : -
Logical Tier Free (GB)                   : -
Logical PRC Total (GB)                   : -
Logical Tier Used (GB)                   : -
Logical FAST SG Usage Total (GB)         : 1
Logical FAST Available (GB)              : -
Logical Max SG Demand Total (GB)        : -
Logical Excess (GB)                      : -
Tier Status : -

Storage Groups (1)
{
-----
      P   FAST SG   Max SG
      r   Usage   Demand
SG Name   Policy   i Log (GB) Log (GB)
-----
SQLServer [OutOfPolicy] 1      1      -
      Total                1      -
}
}

```

## Technology with -vp and -v report description

The verbose report above shows a more detailed break down of the VP tier Totals. The fields in the `Tier` block match the non-verbose report, except that the `ATTR` field is displayed as `Tier Status` and has a full text description that matches the legend.

The following additional fields are displayed in the `Tier` block:

**Logical PRC Total** — Sum of (PRC% \* pool enabled capacity) for all DATA device pools in the tier. The PRC determines how much capacity from each DATA device pool will be reserved for non-FAST activities. If the free space in a given DATA device pool (as a percentage of pool enabled capacity) falls below the PRC then the FAST system may not move any more chunks into this pool.

The totals for the `Tier` block appear in a separate `Technology` block:

**Logical Tier Enabled Total** — Sum of pool enabled capacities of all DATA device pools included in VP tiers that match the technology type. Also the total of tier enabled from the tier block.

**Logical Tier Free Total** — Tier Enabled Total - Tier Used Total, minimum 0.

**Logical Tier Used Total** — Sum of pool allocated capacity for all DATA device pools included in VP tiers that match the technology type. Allocated capacity on all DATA devices will be counted, including DATA devices that are not enabled; therefore Used may be greater than Enabled. Also the total of Tier Used from the Tier block.

**Logical FAST SG Usage Total** — Sum of allocated capacity residing on this technology from thin devices in a storage group associated with a FAST policy containing VP tiers. Also the total of FAST SG Usage from the Tier block.

**Logical FAST Available Total** — Sum of FAST storage group Usage and (tier Free less the PRC from all included DATA device pools, minimum 0) for all VP tiers that match the technology type and are included in associated FAST policies. Also the total of FAST Available from the Tier block.



**Logical Max SG Demand Total** — Sum of the allocated capacity of all thin devices in a storage group associated with a FAST policy containing VP tiers if they were to occupy the full allotted quota (per the limit defined in FAST policy) of space in a VP tier of this technology type. Also the total of **Max SG Demand** from the **Tier** block.

**Logical Excess** — Difference between **FAST Available** and **Max FAST SG Demand Total**. Also the total of **Excess** from the **Tier** block.

The **Storage Group** block shows the following additional information about associated storage groups:

**SG Name** — Storage group name.

**Policy** — Policy name. An **[OutOfPolicy]** entry is shown if devices from the storage group reside on this tier, but the tier is not part of the associated policy.

**Priority** — Priority of the association between the storage group and the policy

**Logical FAST SG Usage** — Sum of allocated capacity from thin devices in the storage group that resides on this VP tier.

**Logical Max SG Demand** — Sum of allocated capacity from thin devices in the storage group if they were to occupy the full allotted quota (as per the FAST policy) of space in the VP tier. If **-allocated** is specified, **Max SG Demand** will be calculated using allocated instead of configured capacity (this value may be less than the **Max SG Demand** column for this storage group and policy in the compliance report if the thin devices are not fully allocated). This value is not applicable if the policy associated with the storage group does not contain the VP tier.

## Listing the technology demand for DP tiers using -v

This example shows the output for a technology demand for DP tiers in verbose mode. Definitions of the output fields are after the display:

```
symfast list -sid 234 -tech FC -demand -dp -v

Symmetrix ID                      : 000194900234

Technology                         : FC
Raw Total(GB)                     :          6000
Raw Free (GB)                     :          1030
Raw Used (GB)                     :          4970
Raw FAST SG Usage Total (GB)      :          4239
Raw FAST Available (GB)           :          5270
Raw Max SG Demand Total (GB)      :          4120
Raw Excess (GB)                   :          +1150

DP Tiers (2)
{
  Tier Name                       : WorkDBTier
  Target Prot                     : R6(6+2)
  Raw Tier Unconfigured (GB)      :          1030
  Logical Tier Configured (GB)    :          1598
  Logical FAST SG Usage Total (GB):          1420
  Logical FAST Available (GB)     :          2193
  Logical Max SG Demand Total (GB):          1440
  Logical Excess (GB)             :          +753
  Tier Status : Tier in a FAST policy associated with SG(s)

Storage Groups (2)
{
```

```

-----
SG Name          FAST Policy      P r FAST SG Usage Log (GB) Max SG Demand Log (GB)
-----
OraMarketing     DBPolicy      1      640      600
OraSales         DBPolicy      2      780      840
-----
Total            1420      1440
}

Tier Name          : WorkTier
Target Prot        : R5(3+1)
Raw Tier Unconfigured (GB) : 1030
Logical Tier Configured (GB) : 1980
Logical FAST SG Usage Total (GB) : 1760
Logical FAST Available (GB) : 2533
Logical Max SG Demand Total (GB) : 1650
Logical Excess (GB) : +833
Tier Status : Tier in a FAST policy associated with SG(s)

Storage Groups (3)
{
-----
SG Name          FAST Policy      P r FAST SG Usage Log (GB) Max SG Demand Log (GB)
-----
Finance2009      FinanceData  1      600      600
Finance2008      FinanceData  2      640      550
Finance2007      FinanceData  2      520      500
-----
Total            1760      1650
}

Tier Name          : [OutOfTier]
Target Prot        : N/A
Raw Tier Unconfigured (GB) : -
Logical Tier Configured (GB) : 150
Logical FAST SG Usage Total (GB) : 0
Logical FAST Available (GB) : 0
Logical Max SG Demand Total (GB) : -
Logical Excess (GB) : -
Tier Status : -

No devices in a FAST SG on tier [OutOfTier]
}

```

The verbose report above shows a detailed break down of the DP tier totals. The fields in the `Tier` block exactly match the non-verbose report, except that the `ATTR` field displays as `Tier Status` and has a full text description that matches the legend.

The `Storage Groups` block shows the following additional information about associated storage groups:

- ◆ `SG Name` — Storage group name.
- ◆ `Policy` — Policy name. An `[OutOfPolicy]` entry is shown if devices from the storage group reside on this tier, but the tier is not part of the associated policy.
- ◆ `Priority` — Priority of the association between the storage group and the policy
- ◆ `Logical FAST SG Usage` — Sum of logical capacities of all standard devices in the storage group with matching RAID protection that reside on this DP tier.

- ◆ **Logical Max SG Demand** — Sum of logical capacities of all standard devices in the storage group if they were to occupy the full allotted quota (as per the FAST policy) of space in the DP tier. This value matches the **Max SG Demand** column for this storage group and policy in the compliance report. This value is not applicable if the policy associated with the storage group does not contain the DP tier.

## Listing the technology demand using V71 mode

When the `-mode v71` option is specified, the `-dp` option is implied, and all fields will revert to their definitions from SE 7.1. The column definitions are listed after the sample command and output:

```
symfast list -sid 234 -tech FC -demand -mode v71
```

```
Symmetrix ID           : 000194900234

Technology              : FC
Total(GB)               :          6000
Free (GB)               :          1030
Used (GB)               :          4970
FAST SG Usage Total (GB) :          4239
FAST Available (GB)     :          5270
Max SG Demand Total (GB) :          4120
Excess (GB)             :          +1150

Tiers (2)
{
  -----
  A                                     Raw Capacities (GB)
  T                                     -----
  T Target      Tier   Tier FAST SG   FAST   Max SG   Excess
  R Prot       Free   Used  Usage   Avail   Demand
  -----
Tier          F R6 (6+2)  1030   2130   1893   2923   1920   +1003
WorkDBTier    F R5 (3+1)  1030   2640   2346   3377   2200   +1177
[OutOfTier]   - N/A      -      200     0      0      -      -
Total                                     -----
                                         4970   4239               4120
}

Legend:
ATTR      : F = Tier in a FAST policy associated with SG(s)
          : P = Tier in a FAST policy unassociated with SG(s)
          : N = Tier not in any FAST policy
```

### Technology Block

- ◆ **Total** — Sum of capacities of all disks that match the technology type. Also the sum of **Free** and **Used**.
- ◆ **Free** — Sum of unconfigured space on all disks that match the technology type.
- ◆ **Used** — Sum of configured space on all disks that match the technology type.
- ◆ **FAST SG Usage Total** — Sum of hyper sizes of all standard devices that reside on this technology and are in a storage group associated with a FAST policy containing DP tiers.

- ◆ **FAST Available — Based on FAST SG Usage Total.** If the `swap_notvisible_devs` control parameter is **ENABLED**, the hyper sizes of not-host-visible (unmapped/unmasked) standard devices that reside on this technology are included in **FAST Available**. If the `allow_only_swap` control parameter is **DISABLED**, **Free** is included in **FAST Available**.
- ◆ **Max SG Demand Total** — Sum of hyper sizes of all Standard devices in a storage group associated with a FAST policy containing standard tiers if they were to occupy the full allotted quota (as per the FAST policy) of space in a standard tier of this technology type. This is an estimated number. Solutions Enabler will try to approximate the physical space that would be occupied by the standard devices based on the tier protection type.

**Excess** — Difference between **FAST Available** and **Max FAST SG Demand**.

## Tier Block

- ◆ **Name** — Shows names of all standard tiers of this technology type.
- ◆ **ATTR** — Shows the status of the standard tier. Tiers can have 1 of 3 possible attributes:
  - In a FAST policy associated with a storage group.
  - In a FAST policy or policies where none of the FAST policies are associated with a storage group.
  - Not in any FAST policy.
- ◆ **Target Prot** — Target protection of the standard tier.
- ◆ **Raw Free** — Unconfigured space in the tier disk groups. If standard tiers overlap, the unconfigured space from disk groups included in both tiers will be attributed to both tiers. Hence the same free space may be counted multiple times. If a disk group does not have enough usable disks to support devices of the tier protection type (disk count of 7 for a RAID-5(7+1) Tier), **Raw Free** capacity for that disk group will be reported as 0.
- ◆ **Raw Used** — Sum of hypers of all standard devices with matching raid protection on this standard tier.
- ◆ **Raw FAST SG Usage** — Sum of the hypers of all standard devices with matching RAID protection that reside on this tier and are in a storage group associated with a FAST policy containing standard tiers.
- ◆ **Raw FAST Available** — If the standard tier is in some FAST policy associated with a storage group, the **FAST Available** capacity is the sum of **Raw FAST SG Usage**, **Raw Free**, and sum of hyper sizes of not visible standard devices (unmapped/unmasked) that reside on the tier. Free space will be added only if the `allow_only_swap` control parameter is disabled. Space occupied by not visible standard devices will be added only if the `swap_notvisible_devs` control parameter is enabled. If the tier is not in any FAST policy or is in policies where none of the policies are associated to a storage group then the **FAST Available** capacity is same as **FAST SG Usage**.
- ◆ **Raw Max SG Demand** — Sum of hypers of all standard devices in a Storage Group associated with a FAST Policy containing this Standard Tier if the devices were to occupy the full allotted quota (as per the FAST Policy) of space in the Standard Tier. This is an estimated number. Solutions Enabler will try to approximate the physical

space that would be occupied by the Standard Devices based on the Tier protection type. If the Tier is not in any FAST Policy or is in Policies where none of the Policies are associated to a Storage Group then this value is Not applicable.

- ◆ `Raw Excess` — Difference between Raw FAST Available and Raw Max FAST SG Demand. If the Standard Tier is not in any FAST Policy or is in Policies where none of the Policies are associated to a Storage Group then this value is Not applicable.

---

**Note: 1** If on a given technology there exist standard devices that do not belong to any standard tier they will be shown as `[OutOfTier]`. This can happen when the RAID type of standard devices does not match any tier RAID type, or when the defined standard tiers only contain a subset of the disk groups of the technology.

---



# APPENDIX D

## Virtual LUN Migration Example

This appendix provides an example of a Symmetrix Virtual LUN migration.

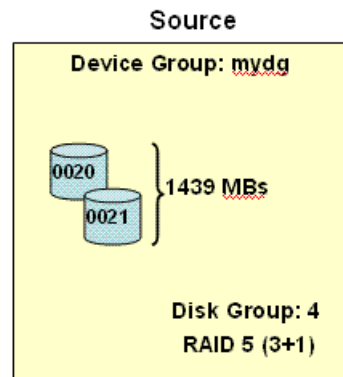
- ◆ Virtual LUN migration example ..... 456

## Virtual LUN migration example

This section provides a step-by-step `symmigrate` example. It walks through each step that is involved in a single migration session.

In this example, a device group will be used as the source. The source devices are RAID5 (3+1) devices that are specified as a TGT list of devices in a device group named `mydg`.

Device group `mydg` has the following TGT devices: 0020, 0021 with 1439 MBs in size and the protection type is RAID5 (3+1). The disk group number is 4.



**Figure 37** Device view prior to migration session

The following `validate` command will validate the `symmigrate` request and generate the device to device pairs in the output file named `mydgtosgout`:

```
symmigrate validate -nop -v -outfile mydgtosgout
  -g mydg -tgt
  -tgt_raid6 -protection 6+2 -tgt_dsk_grp 1
  -tgt_config
  -name myfullsession
```

'Validate' operation execution is in progress for the device group - mydg. Please wait...

STARTING a Migrate 'Validate' operation.

SELECTING the list of Source devices

Devices 0020, 0021 [SELECTED]

The Migrate 'Validate' operation SUCCEEDED.

'Validate' operation successfully executed for the device group - mydg.

The following command shows the validated device-to-device pairs in the output file:

```
more mydgtosgout
```

```
0020 0075
0021 0086
```



After the `validate` command, use the `establish` command to establish the migration session by using the validated output file `mydgtosgout`.

```
symmigrate establish -sid 123 -file mydgtosgout -nop -v -name myfullsession
```

'Establish' operation execution is in progress for the device list in device file 'mydgtosgout'. Please wait...

STARTING a Migrate 'Establish' operation.

PAIRING of Source and Target Devices:

Devices: 0020-0075, 0021-0086

The Migrate 'Establish' operation SUCCEEDED.

'Establish' operation successfully executed for the device list in device file 'mydgtosgout'.

For the preceding command, the following audit log messages will be generated:

STARTING a Migrate 'ESTABLISH' operation for Device Pairs. Options=(RAID6\_6+2, DiskGroup 1, Configured Space)

Symm 000192600123 Source Devices: [0020 0021] Target Devices: [0075 0086]

The Migrate 'ESTABLISH' operation SUCCESSFULLY COMPLETED.

The following `symmigrate list` command will list all the existing migration sessions on this Symmetrix array:

```
symmigrate list -sid 123
```

Symmetrix ID: 000192600123

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
0010	N/A	0	Migrated	100	Migrate1
0014	N/A	0	Migrated	100	Migrate1
0020	0075	23024	SyncInProg	50	myfullsession
0021	0086	32209	SyncInProg	30	myfullsession
Total -----					
Tracks		55233			
MB(s)		1726.0			

After the `list` command, use the `symmigrate query` command with the `-detail` option to query the `myfullsession` migration session. This provides more detailed information about this session:

```
symmigrate query -name myfullsession -sid 123 -detail
```

Symmetrix ID: 000192600123

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	TGT Dsk Grp	TGT Protection	Session Name
0020	0075	18420	SyncInProg	60	01 RAID-6 (6+2)	myfullsession	
0021	0086	27628	SyncInProg	40	01 RAID-6 (6+2)	myfullsession	
Total -----							
Tracks		46048					

MB(s) 1439.0

The `symmigrate query` command with the `-i` option can also be used to monitor the progress of this migration session:

**`symmigrate query -name myfullsession -sid 123 -i 5 -c 2`**

Symmetrix ID: 000192600123

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
0020	0075	13814	SyncInProgress	70	myfullsession
0021	0086	23024	SyncInProgress	50	myfullsession

Total -----  
 Tracks 36834  
 MB(s) 1151.1

Copy rate : 4.7 MB/S  
 Estimated time to completion : 00:02:37

Symmetrix ID: 000192600123

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
0020	0075	13010	SyncInProgress	72	myfullsession
0021	0086	22310	SyncInProgress	52	myfullsession

Total -----  
 Tracks 35320  
 MB(s) 1103.8

Copy rate : 4.7 MB/S  
 Estimated time to completion : 00:02:30

As the migration session is in `SyncInProgress` state, the following `symdev show` command will display the device RAID group information for both the source and the target devices. Both the primary and the secondary RAID group will be shown for the source device and only the original primary RAID group will be shown for the target device.

The following is the `device show` output for source device 0020:

**`symdev show 0020 -sid 123`**

```

Device Physical Name      : Not Visible

Device Symmetrix Name     : 0020
.
.
.
Device Configuration     : RAID-5              (Non-Exclusive Access)

Device is WORM Enabled    : No
Device is WORM Protected : No

SCSI-3 Persistent Reserve: Disabled

Dynamic Spare Invoked     : No

Dynamic RDF Capability     : RDF1_OR_RDF2_Capable

STAR Mode                 : No

```

```

STAR Recovery Capability : None
STAR Recovery State      : NA

Device Service State     : Normal

Device Status            : Ready          (RW)
Device SA Status         : N/A            (N/A)

Mirror Set Type          : [RAID-5,RAID-6,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              : RAID-5
  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              : RAID-6
  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
}

```

## RAID Group Information

```

{
  Mirror Number           : 1
  RAID Type               : RAID-5
  Protection Level        : 3+1
  Device Position         : Primary
  RAID Group Service State : Normal
  Failing Member Mask     : N/A
  Hyper Devices:
  {
    Device : 0020
    {
      -----
      Disk      DA      Hyper      Member      Disk
      DA :IT    Vol#    Num Cap(MB)  Num Status  Grp#  Cap(MB)
      -----
      16B:Dd    841     18   2878    1 RW        4   140014
      01C:D9    720     18   2878    2 RW        4   140014
      16D:C4    137     18   2878    3 RW        4   140014
      01A:Ce    460     18   2878    4 RW        4   140014
    }
  }
}

```

```

Mirror Number           : 2
RAID Type               : RAID-6
Protection Level        : 6+2
Device Position         : Secondary
RAID Group Service State : Normal
Failing Member Mask     : N/A
Hyper Devices:
{
  Device : 0020
  {
    -----
    Disk      DA      Hyper      Member      Disk

```

```

        DA :IT    Vol#    Num Cap(MB)    Num Status    Grp#    Cap(MB)
        -----
        01A:C4    176     55    1439    1 RW          1    140014
        16B:D7    674     55    1439    2 RW          1    140014
        16C:D6    678     55    1439    3 RW          1    140014
        01D:C7    224     55    1439    4 RW          1    140014
        16A:D6    682     55    1439    5 RW          1    140014
        01B:C7    244     55    1439    6 RW          1    140014
        01C:C8    275     54    1439    7 RW          1    140014
        16D:Dd    861     54    1439    8 RW          1    140014
    }
}
}

```

The following is the device show output for target device 0075:

#### **symdev show 0075 -sid 123**

```

Device Physical Name      : Not Visible

Device Symmetrix Name     : 0075
.
.
.
Device Configuration     : RAID-6              (Non-Exclusive Access)

Device is WORM Enabled    : No
Device is WORM Protected : No

SCSI-3 Persistent Reserve: Disabled

Dynamic Spare Invoked     : No

Dynamic RDF Capability     : RDF1_OR_RDF2_Capable

STAR Mode                 : No
STAR Recovery Capability  : None
STAR Recovery State       : NA

Device Service State      : Normal

Device Status             : Not Ready          (NR)
Device SA Status          : N/A                (N/A)

Mirror Set Type           : [RAID-6,N/A,N/A,N/A]

Mirror Set DA Status      : [NR,N/A,N/A,N/A]

Mirror Set Inv. Tracks    : [0,0,0,0]

Back End Disk Director Information
{
    Hyper Type                : RAID-6
    Hyper Status               : Not Ready          (NR)
    Disk [Director, Interface, TID] : [N/A,N/A,N/A]
    Disk Director Volume Number : N/A
    Hyper Number               : N/A
    Mirror Number              : 1
}
RAID Group Information
{
    RAID Type                  : RAID-6
    Device Position            : Primary
    RAID Group Number          : 361
    RAID Group Capacity (MB)   : 8631
    RAID Group Ready State     : Not Ready
    RAID Group WriteProtect State : Enabled
}

```

```

Failing Member Mask          : N/A
RAID-6 Hyper Devices (6+2):
{
  Device : 0FA8
  {
    -----
    Disk      DA      Hyper      Member      Disk
    DA :IT    Vol#    Num Cap(MB) Num Status  Grp#  Cap(MB)
    -----
    01A:C4    176     55    1439    1 NR      1    140014
    16A:D6    682     55    1439    5 NR      1    140014
    01B:C7    244     55    1439    6 NR      1    140014
    16B:D7    674     55    1439    2 NR      1    140014
    01C:C8    275     54    1439    7 NR      1    140014
    16C:D6    678     55    1439    3 NR      1    140014
    01D:C7    224     55    1439    4 NR      1    140014
    16D:Dd    861     54    1439    8 NR      1    140014
  }
}

```

The following `symmigrate query` command shows that the migration operation is finished and the migration session is in the Migrated state:

```
symmigrate query -name myfullsession -sid 123
```

```
Symmetrix ID: 000192600123
```

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
0020	0075	0	Migrated	100	myfullsession
0021	0086	0	Migrated	100	myfullsession
Total -----					
Tracks		0			
MB(s)		0			

When the migration session is in the Migrated state, use the `symmigrate terminate` command to remove the session, as shown:

```
symmigrate terminate -sid 123 -name myfullsession -nop
```

```
'Terminate' operation execution is in progress for migration session
- myfullsession. Please wait...
```

```
'Terminate' operation successfully executed for migration session
- myfullsession.
```

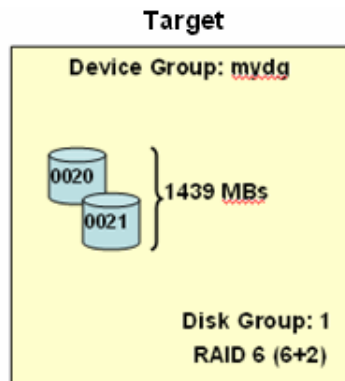
For the preceding command, the following audit log messages are generated:

```
STARTING a Migrate 'TERMINATE' operation for Session Name. Options=(None)
```

```
Symm 000192600123 Session Name: [myfullsession]
```

```
The Migrate 'TERMINATE' operations SUCCESSFULLY COMPLETED.
```

When the migration is complete, the devices look as shown in [Figure 38](#)



**Figure 38** Device view after the migration session

The following is the output for `symmigrate list` command after the `terminate` command (`myfullsession` no longer displays):

```
symmigrate list -sid 123
```

Symmetrix ID: 000192600123

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
0010	N/A	0	Migrated	100	Migrate1
0014	N/A	0	Migrated	100	Migrate1
Total -----					
Tracks		0			
MB(s)		0			

The following `symdev show` command displays the RAID group information for both the source and the target device after the migration has finished:

```
symdev show 0020 -sid 123
```

```

Device Physical Name      : Not Visible
Device Symmetrix Name     : 0020
.
.
.
Device Configuration     : RAID-6              (Non-Exclusive Access)

Device is WORM Enabled    : No
Device is WORM Protected : No

SCSI-3 Persistent Reserve: Disabled

Dynamic Spare Invoked     : No

Dynamic RDF Capability     : RDF1_OR_RDF2_Capable

STAR Mode                 : No
STAR Recovery Capability   : None
STAR Recovery State       : NA

Device Service State      : Normal

Device Status             : Ready              (RW)

```

Device SA Status : N/A (N/A)

Mirror Set Type : [RAID-6,N/A,N/A,N/A]

Mirror Set DA Status : [RW,N/A,N/A,N/A]

Mirror Set Inv. Tracks : [0,0,0,0]

#### Back End Disk Director Information

```
{
  Hyper Type           : RAID-6
  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
}
```

#### RAID Group Information

```
{
  Mirror Number        : 1
  RAID Type            : RAID-6
  Device Position      : Primary
  RAID Group Number    : 361
  RAID Group Capacity (MB) : 8631
  RAID Group Ready State : Not Ready
  RAID Group WriteProtect State : Enabled
  Failing Member Mask  : N/A
  RAID-6 Hyper Devices (6+2):
```

```
{
  Device : 0FA8
  {
    -----
    Disk      DA      Hyper      Member      Disk
    DA :IT    Vol#    Num Cap(MB)  Num Status  Grp#  Cap(MB)
    -----
    01A:C4    176     55    1439    1 RW        1    140014
    16A:D6    682     55    1439    5 RW        1    140014
    01B:C7    244     55    1439    6 RW        1    140014
    16B:D7    674     55    1439    2 RW        1    140014
    01C:C8    275     54    1439    7 RW        1    140014
    16C:D6    678     55    1439    3 RW        1    140014
    01D:C7    224     55    1439    4 RW        1    140014
    16D:Dd    861     54    1439    8 RW        1    140014
  }
}
```

```
}
}
```

#### # symdev show 0075 -sid 123

Device Physical Name : Not Visible

Device Symmetrix Name : 0075

.  
.  
.

Device Configuration : RAID-5 (Non-Exclusive Access)

Device is WORM Enabled : No

Device is WORM Protected : No

SCSI-3 Persistent Reserve: Disabled

Dynamic Spare Invoked : No

Dynamic RDF Capability : RDF1\_OR\_RDF2\_Capable

```

STAR Mode                : No
STAR Recovery Capability : None
STAR Recovery State      : NA

Device Service State     : Normal

Device Status            : Ready                (RW)
Device SA Status         : N/A                  (N/A)

Mirror Set Type          : [RAID-5,N/A,N/A,N/A]

Mirror Set DA Status     : [RW,N/A,N/A,N/A]

Mirror Set Inv. Tracks   : [0,0,0,0]

Back End Disk Director Information
{
  Hyper Type              : RAID-5
  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
}

RAID Group Information
{
  Mirror Number           : 1
  RAID Type               : RAID-5
  Device Position         : Primary
  RAID Group Number       : 122
  RAID Group Capacity (MB) : 8631
  RAID Group Ready State  : Ready
  RAID Group WriteProtect State : Enabled
  Failing Member Mask     : N/A
  RAID-5 Hyper Devices (3+1):
  {
    Device : 05B1
    {
      -----
      Disk      DA      Hyper      Member      Disk
      DA :IT    Vol#    Num Cap(MB)  Num Status  Grp#   Cap(MB)
      -----
      01A:Ce    460     18   2878     4 RW        4   140014
      16B:Dd    841     18   2878     1 RW        4   140014
      01C:D9    720     18   2878     2 RW        4   140014
      16D:C4    137     18   2878     3 RW        4   140014
    }
  }
}

```



# APPENDIX E

## Updating the Host

After you reconfigure a Symmetrix array by moving, deleting, adding, or modifying one or more devices, you must update the host so that the host recognizes the new Symmetrix configuration.

For some platforms, the `symcfg scan` command is available to perform the host update. These include Sun Solaris, HP-UX, IBM AIX, Tru64/OSF1, Windows, and HP Open VMS systems.

This appendix describes the procedures for these host systems:

- ◆ Sun Solaris systems ..... 466
- ◆ HP-UX systems ..... 466
- ◆ IBM AIX systems ..... 466
- ◆ HP Tru64 UNIX systems ..... 467
- ◆ Windows systems ..... 467
- ◆ HP Open VMS systems ..... 468

## Sun Solaris systems

Whenever devices are newly mapped to a host system, or device channel addresses are changed, you need to perform the following actions to introduce the new devices to the system.

To add new Symmetrix devices while online in the Solaris environment, perform the following steps:

1. Execute the following utilities:

```
drvconfig or symcfg scan  
disks  
devlinks
```

2. Follow the instructions in the Solaris documentation to introduce new devices to the host environment.

## HP-UX systems

Whenever devices are newly mapped to a host system or device channel addresses are changed, you need to perform the following actions to introduce the new devices to the system.

To view the results of the mapping changes, use the `ioscan` command in a statement similar to the following:

```
ioscan -fnC disk
```

You can define newly connected physical volumes to the HP-UX host system without rebooting it, using the following form:

```
insf -e
```

For more information, refer to the HP 9000 documentation.

## IBM AIX systems

Whenever devices are newly mapped to a host system or device channel addresses are changed, you need to perform the following actions to introduce the new devices to the AIX system:

1. From the **SMIT** menu, select **Devices > Fixed Disk > Add a Disk**.
2. Select the **EMC SYMMETRIX** definition from the disk table.
3. Select the SCSI bus on which the new disk resides.
4. Type the connection address for the new device (target, LUN).
5. Select **EXECUTE**.
6. Repeat steps 2 through 5 for each new device being added to the configuration.

## HP Tru64 UNIX systems

Whenever devices are newly mapped to a host system or device channel addresses are changed, you need to perform the following actions to introduce the new devices to the system.

When using Tru64 UNIX, you can introduce the new devices to the system with these steps:

1. At the prompt, type:

```
scsimgr -scan_bus bus=BUSNUM
```

2. Repeat for each LUN:

- a. Write a label to the device you are defining:

```
disklabel -rw rz<lun_letter><unitID> <label>
```

- b. Change the ownership on the device to a particular application:

```
chown <owner>:<group> *rz<lun letter><unitID>*
```

3. Follow the host documentation to introduce new devices to the host environment.

## Windows systems

Whenever devices are newly mapped to a host system or device channel addresses are changed, you need to perform the following actions to introduce the new devices to the system.

### Windows 2000

To add new Symmetrix devices to a Windows 2000 system while the system remains online, perform these steps:

1. From the desktop, select **Start, Settings, Control Panel, Add /Remove Hardware**. Complete the wizard to discover and add the new Symmetrix devices.
2. Partition and format the new devices as described in the *Windows 2000* chapter of the *Open Systems Host Environment Product Guide*.

### Windows 2003

To add new Symmetrix devices to a Windows 2003 system while the system remains online, perform these steps:

1. From the desktop, select **Start, Settings, Control Panel, Add /Remove Hardware**. Complete the wizard to discover and add the new Symmetrix devices.
2. Partition and format the new devices as described in the Windows 2003 documentation.

## HP Open VMS systems

Whenever devices are newly mapped to an OpenVMS system, or device paths or addresses are changed, the following actions must be performed. If a VMS identifier (device address) is changed on an existing device, the device must be dismounted before performing this step.

To make the results of the changes visible to the host system, use the SYSMAN utility, as follows:

```
$ mcr sysman
SYSMAN> io scsi_path_verify
SYSMAN> io autoconfig
SYSMAN> exit
```

# APPENDIX F

## SIU: Overview and Management

This chapter describes the Symmetrix Integration Utilities (SIU) disk management operations for your storage system in a Windows Server environment.

Topics include:

- ◆ Introduction ..... 470
- ◆ Configuring a VMware virtual server ..... 472
- ◆ Viewing detailed information about disks..... 474
- ◆ Mounting and unmounting disks..... 475
- ◆ Manipulating disk signatures ..... 477
- ◆ Updating partition tables ..... 477
- ◆ Flushing cache buffers to disk..... 478
- ◆ Device masking operations ..... 478
- ◆ Using volume flags..... 479
- ◆ Performing typical storage operations ..... 480
- ◆ SIU examples ..... 485
- ◆ XML examples..... 486

# Introduction

The Symmetrix Integration Utilities (SIU) integrates and extends the Windows 2003 and Windows 2008 disk management functionality to better operate with and on TimeFinder and Symmetrix Remote Data Facilities (SRDF) devices.

SIU operations integrate with the management functionality of TimeFinder and SRDF with support specifically for Windows servers by extending their ability to:

- ◆ Mount and unmount hard disks and their associated file systems.
- ◆ Flush file system cache buffers to disk.
- ◆ Manipulate disk signatures.

SIU is not a replacement for the Windows Logical Disk Manager (LDM), but bridges these lacking functionalities necessary for Windows users to optimally work with EMC business continuance storage devices.

---

**Note:** Windows 2008 Server systems need the Microsoft hotfix 952790 installed. (<http://support/microsoft.com/kb/952790>).

---

## SIU capabilities

SIU enables you to perform the following actions:

- ◆ View the physical disk, volume, and VMware datastore configuration data.
- ◆ Update the partition table on a disk.
- ◆ Set and clear volume flags.
- ◆ Flush any pending cached file system data to disk.
- ◆ Show individual disk, volume, or VMware datastore details.
- ◆ Mount and unmount volumes to a drive letter or mount point.
- ◆ Manipulate disk signatures.
- ◆ Scan the drive connections and discovers any new disks available to the system.
- ◆ Mask devices to and unmask devices from the Windows host.

## Privilege requirements

SIU is based on the functionality and features of associated hardware and software in an EMC Enterprise Storage environment managed and used on Windows 2003 and Windows 2008 platforms. Most command actions require administrative privileges. On Windows 2008, you must be a local administrator, or the command prompt must be `Run as administrator`.

---

**Note:** For security reasons, Windows Server 2008 does not honor the `Start In` path property of the shortcut `Run as administrator`. Therefore, the shortcut will always begin in the Windows system directory, instead of the SIU directory.

---

## SIU commands

The SIU CLI commands are a syntactical set and form of one command (`symntctl`), which has various arguments to support each of the main functional operations or actions. A summary of the SIU command set that operate in a Windows operating system environment is listed in [Table 30](#). These commands can be used in scripts when working on Windows 2003 or Windows 2008 platforms

**Table 30** SIU command summary

Command	Argument	Action
symntctl		Performs disk management operations for Windows platforms to properly work with Business Continuance devices. Use with the following arguments for other operations.
	flag	Sets and clears volume flags.
	flush	Flushes all pending unwritten file system cache entries to disk.
	list	Lists all visible physical disks, volumes, and VMware datastores.
	mask	Removes access to the specified Symmetrix device on all HBAs of the host where the specified device is visible.
	mount	Mounts the specified volume to the specified mount point or drive letter.
	rescan	Scans the device connections and discovers any new disks available to the system.
	show	Displays individual disk, volume, or VMware datastore details.
	signature	Manipulates disk signatures.
	umount	Unmounts a volume from the drive letter and all mount points.
	unmask	Adds access to the specified Symmetrix device on all HBAs of the host that is connected to the specified Symmetrix array.
	update	Updates the partition table on a disk.

For command syntax information, refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference*.

## Volume management software

Windows Server 2003 and Windows Server 2008 have two types of disks: basic and dynamic. Basic disks support primary partitions only, while dynamic disks support dynamic volumes only.

The Windows Logical Disk Manager (LDM) provides full management functionality for basic disks, but only limited management functionality for dynamic disks. Most of the advanced capabilities of dynamic disks are only available with Veritas Storage Foundation for Windows (SFW).

## SFW dynamic disk groups with TimeFinder

For Windows advanced server systems with dynamic disks, SIU functionality works with Veritas Storage Foundation for Windows.

If you are going to set up and use Veritas dynamic disk groups (VDGs) on a production server, you must comply with the following requirements when configuring VDGs for use with Solutions Enabler TimeFinder/Mirror software:

- ◆ Create a TimeFinder/Mirror device group for every dynamic disk group, maintaining the one-to-one basis. To keep this relationship clear, name the dynamic disk groups the same as the TimeFinder/Mirror device groups.
- ◆ It is not necessary to create the disk group on the backup server, since the Logical Disk Manager (LDM) database is resident on every dynamic disk, as it is copied over to the backup server on each of the BCVs. All disk group definitions are stored in this database and will immediately appear on the backup server (in an offline state). You can create the TimeFinder/Mirror device groups on either the production server or the backup server.

Be aware of the following when creating TimeFinder/Mirror device groups:

- Do not mix STD and BCV devices in the same VDG.
- Do not have Symmetrix and non-Symmetrix storage type devices in the same VDG.
- Do not mix R1 and R2 devices in the same VDG.
- Do not add gatekeeper devices to any VDG.

---

**Note:** SFW 5.1 on windows server 2008 is not supported.

---

## Configuring a VMware virtual server

When the SIU server is on a VMware virtual server (VM), there are additional setup requirements and steps:

- ◆ Enable the SPC-2 flag on the Symmetrix ports being used.
  - ◆ Set the environment variable VICLIENT\_DISABLE\_CACHE to 1.
1. Install VMware Tools on the backup/recovery server.
  2. Open a command prompt to the Solutions Enabler `shlib` directory which is typically located at:
 

```
C:\Program Files\EMC\SYMCLI\shlib
```
  3. Run the following command, `vicfg setup`. This utility is designed to configure access to the ESX Server by enabling SSL ports on the firewall, configuring certificates, and encrypting and storing the login credentials:
    - When prompted, please enter the username and password to be used as credentials for the ESX Server. These credentials are encrypted and stored in the Solutions Enabler authorization database, so there is no concern over their security.
  4. Assign at least 2 Symmetrix gatekeeper devices to the VM. These gatekeepers will be required in order for Solutions Enabler to discover the Symmetrix from the VM:
    - Verify that the proper zoning is in place to connect the HBAs in the ESX Server to the Symmetrix.
    - Using Symmetrix device masking (`symmask.exe`), assign access for the ESX Server to the gatekeeper devices.
    - Configure the VM to have Raw Device Mappings (RDM) for each of the gatekeeper devices. The VM should now see the gatekeepers in a SCSI inquiry (`syminq.exe`).



5. Run the command `symcfg discover` to verify that Solutions Enabler can discover the Symmetrix array.

SIU does not automatically interoperate with the VMware high availability software such as Vmotion, or the VMware load-balancing software such as Distributed Resource Scheduler (DRS). If the VM moves to a different ESX Server, whether through manual intervention or through one of these products, the configuration and authorization needs to be performed again.

---

**Note:** The Symmetrix Integration Utilities (SIU) supports VMware ESX Server V4.0 and higher. However, to facilitate integration into the VMware environment, you must disable all caching by setting the environment variable `VICLIENT_DISABLE_CACHE = 1`.

---

## Configuring the ESX 3.5 Server

Follow these steps to configure the ESX 3.5 Server:

---

**Note:** Make sure you are using VMware ESX Server 3.5 Update 2.

---

1. Install VMware Tools on the backup/recovery server.
2. Open a command prompt to the Solutions Enabler `shlib` directory which is typically located at:

```
C:\Program Files\EMC\SYMCLI\shlib
```

3. Run the following command:

```
vicfg setup
```

This utility is designed to configure access to the ESX Server by enabling SSL ports on the firewall, configuring certificates, and encrypting and storing the login credentials:

4. When prompted, enter the username and password to be used as credentials for the ESX Server. These credentials are encrypted and stored in the Solutions Enabler authorization database for security.

## Configuring the ESX 4.0 Server

Follow these steps to configure the ESX 4.0 Server:

1. Setup the vSphere Management appliance, which serves as CIM server, as follows:
  - a. Open the VMware vSphere Client.
  - b. Select from the menu options: **Open >Deploy OVF Template >Deploy from URL**. Provide the following URL to setup vSphere Management appliance:

```
http://www.vmware.com/go/importvma/vma4.ovf
```

This creates a virtual machine which serves as a CIM Server.

- c. When prompted, enter the username and password (must have root privileges). For example: **vi-admin**
  - d. Start the VM and run the following scripts:

```
sudo /etc/vmware/vmware-smis-setup.pl
```

- e. Run following commands to open the firewall on unsecured port 5988:

```
sudo iptables -A INPUT -p tcp --dport 5988 -j ACCEPT
sudo /etc/init.d/iptables save
sudo /etc/init.d/iptables restart
```

- f. Make sure that all iptable entries are proper.

2. Install VMware tools on the backup/recovery server.

The Virtual Machine needs Fully Qualified Domain Name (FQDN). For example:

hostname.domain.com. -note: trailing '.'

---

**Note:** The Virtual Machine needs Fully Qualified Domain Name(FQDN). For example:  
hostname.domain.com. -note:trailing '.'

---

3. Open a command prompt to the Solutions Enabler `shlib` directory which is typically located at:

```
C:\Program Files\EMC\SYMCLI\shlib
```

4. Run the following command:

```
vicfg setup
```

The utility is designed to configure access to the ESX Server by enabling SSL ports on the firewall, configuring certificates, and encrypting and storing the login credentials.

5. When prompted, enter the username and password to be used as credentials for the ESX Server. These credentials are encrypted and stored in the Solution Enabler authorization database for security.
6. An entry in the Solution Enabler authorization database pointing to CIM server should be made by running the following command:

```
symcfg auth add -host <HostName> -username <UserName> -password <PassWord>
               -namespace vmware/cim -port 5988 -vmware
```

The output of the `symcfg auth list -vmware` command should be similar to the following:

Hostname	Username	Namespace	Port
-----	-----	-----	-----
10.31.96.111	root	vmware/esxv2	5988
10.31.99.214	vi-admin	vmware/cim	5988

## Viewing detailed information about disks

The SIU list action provides a consolidated view for displaying and managing physical disk, volume, and VMware datastore configuration data.

To view disk numbers in your storage configuration, enter:

```
symmtctl list -disk
```

To view the volume configuration, enter:

```
symmtctl list -volume
```

To view VMware datastores in the virtual infrastructure, enter:

```
symmctl list -datastore
```

The `show` action provides detailed information about an individual disk, volume, or VMware datastore.

To target a specific disk enter:

```
symmctl show -pd diskN
```

where *diskN* is the disk number.

To target a specific volume enter:

```
symmctl show -vol VolName [-g VolGroup]
```

To target a specific VMware datastore enter:

```
symmctl show -datastore DatastoreName
```

## Specifying the output type

SIU allows you to specify the output format for the `symmctl list` and `symmctl show` commands, as follows:

- ◆ XML\_ELEMENT — Provides the output using individual XML elements.
- ◆ XML\_ATTRIBUTE — Provides the output using concatenated XML attributes.

Samples of XML output are provided in [“XML examples” on page 486](#).

## Mounting and unmounting disks

After splitting BCV or SRDF volumes, you can reassign the drive letters for those volumes using the SIU `mount` action. The SIU `mount` action lets you mount hard disks and their associated file systems in your storage environment.

The `mount` action mounts the specified volume name, GUID, or Symmetrix device to the specified drive letter or mount point:

```
symmctl mount -drive DriveLetter | -path MountPnt  
-vol VolName [-g VolGroup] | -guid VolGuid |  
-sid SymId | -symdev SymDev | -part Partition#
```

SIU utilizes Microsoft Virtual Disk Service (VDS) functionality to make a volume permanently `dismounted` (offline) while performing unmount operations. Other applications are prevented from accessing the volumes in this state, thus ensuring data integrity during TimeFinder/Mirror establish and split operations. The volume can be brought back online with the SIU `mount` operation.

## SIU and SYMAPI database

When performing `mount` operations on the Symmetrix device number (`-symdev` option), SIU relies on the SYMAPI database to obtain physical drive information. The physical drive information within the SYMAPI database is created and updated following a Symmetrix `discover` command `symcfg discover`. Therefore, if any device changes are made to the mount host, it is recommended to run the `symcfg discover` command to update the SYMAPI database to reflect those changes. If device changes are made frequently to the mount host, then the `discover` should be scripted as a part of the mount process.

When `symntctl mount` or `umount` commands are called, SIU takes an exclusive lock against the SYMAPI database. If other Solutions Enabler commands occur in parallel to the SIU operations, SIU waits on any locks against the SYMAPI database and continues after the locks are released.

Because SIU `mount` and `umount` operations can interact with the SYMAPI database, it is recommended that at least two gatekeeper devices are presented to the SIU mount host.

## Disk unmount operations

On the BCV or SRDF target (R2) side, establishing a BCV with its standard volume makes the BCV unavailable to the Windows system. Windows systems generate error messages if a mounted volume is referenced while unavailable.

The SIU `umount` action provides the ability to unmount hard disks and their associated file systems, and also removes the drive letter from the specified disk volume.

You can target a drive letter or mount point:

```
symntctl umount -drive DriveLetter
symntctl umount -path MountPnt
```

Or, you can target a volume or global ID:

```
symntctl umount -vol VolName [-g VolGroup]
symntctl umount -guid VolGuid
```

Or, you can target a Symmetrix device or physical device, or device partition:

```
symntctl umount -sid SymmID -symdev SymDev [-part Partition#]
symntctl umount -pd Pdev [-part Partition#]
```

The `-force` option can be used when unmounting to ignore and break any handles open to the volume by other applications.

Unmount operations affect the current drive letter mapping. The command performs a file system flush as part of the dismount process, and the drive is flagged as permanently dismounted (offline) until a subsequent `mount` operation. This ensures that no other applications can access the volume while it is being unmounted, thus ensuring data integrity during TimeFinder/Mirror establish and split operations.

When an unmount operation is performed, SIU and VDS does the following:

1. Gets a handle to the volume being unmounted from the operating system.
2. Obtains an exclusive lock on that volume from the operating system (unless `-force` is specified).
3. Flushes all pending file system cached data to disk for that volume.
4. Dismounts the volume by calling `DeviceIoControl()` with the `FSCTL_DISMOUNT_VOLUME` control code.
5. Offlines the volume by calling `DeviceIoControl()` with the `IOCTL_VOLUME_OFFLINE` control code.

---

**Note:** Offlining of volumes on dynamic disks is not supported on Windows 2003 Server. Therefore, the unmounting operation can not be performed on dynamic disks on Windows 2003 Server.

---

6. Deletes all drive letter or mount points associated with the volume and releases the lock.

## Manipulating disk signatures

In the Windows operating system, disks are identified by disk signatures (also known as labels).

---

**Note:** In Windows Server 2003/2008, GPT (GUID Partitioned) disks are identified by disk ID. The disk signature operation is not applicable to GPT disks.

---

Using a signature action, you can assign, change, initialize, and erase a signature from a disk.

Specify the physical disk for a signature operation using the `-pd` option, or use the Symmetrix ID and Symmetrix device options:

```
symmctl signature -pd diskN ...
```

where *diskN* is the physical disk number.

or

```
symmctl signature -sid SymmID -symdev SymDev
```

To view disk numbers in your storage configuration, enter:

```
symmctl list -signature
```

Using the `-sig` option, the signature of a disk is assigned/changed to any desired label specified in this command option.

Using the `-initialize` option, SIU assigns signatures to any disks discovered without signatures.

Using the `-erase` option, the signature of the specified disk is erased. This option is only used when setting up shared disks and TimeFinder configurations.

---

**Note:** With Windows 2003/2008, this option will erase the signature, only to have Windows immediately re-generate a new signature. Therefore, the volume will appear with a different signature, instead of an empty signature.

---

## Updating partition tables

SIU allows the Symmetrix device partition tables to be updated using the `symmctl update` command with the following options:

- ◆ Use the `-all` option to update the partition tables of each physical disk.
- ◆ Use the `-pd PhysicalDrive#` option to update the partition table for the specified physical drive, for example:

```
symmctl update -pd \\.\PhysicalDrive#
```

- ◆ Use the Symmetrix ID and device number to update the partition table for the specified Symmetrix device, for example:

```
symmtctl update -sid 309 -symdev 01e3
```

## Flushing cache buffers to disk

TimeFinder/Mirror operations such as an on-the-fly BCV split require that the file system cache buffers be flushed to disk for a clean and complete split. The SIU flush action does this without removing the drive letter or volume association. This flush action is needed where the standard/BCV or SRDF R1/R2 pair is split without taking down an application.

You can target the standard or R1 device to flush with a drive letter or a mount point:

```
symmtctl flush -drive DriveLetter
symmtctl flush -path MountPnt
```

Or, you can target the standard device to flush with a volume name or global ID (GUID):

```
symmtctl flush -vol VolName -g VolGroup
symmtctl flush -guid VolGUID
```

Because this command only flushes the Windows file system, applications that handle their own cache management (such as SQL Server or Exchange) are not affected. Flush or checkpoint mechanisms embedded in the application must flush its own application data. The application data must be flushed first, followed by a file system flush.

For example, to flush a volume with the volume GUID of

Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}, enter:

```
symmtctl flush -guid Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}
```

## Device masking operations

It is often necessary to add and remove access to a given Symmetrix device for the current host. This is known as *device masking*. Device masking typically requires several steps to make the device visible or hidden to the host, especially if multiple HBAs and PowerPath are being used. SIU provides the ability to perform device masking operations with a single command by leveraging some of its internal functionality, like system rescan. When using SIU to mask or unmask a device, SIU does the following:

1. Obtains a list of all of the HBAs connected to the specified Symmetrix array on the current host.
2. Determines the current state of the specified device on each of the HBAs (whether it is currently masked or unmasked).
3. Either adds or removes access to the specified device (depends on whether it is a mask or unmask operation) for each of the HBAs.
4. Performs a system rescan of the host so that the newly masked devices will be removed from the host, and the newly unmasked devices will appear on the host.

For SIU device masking to work correctly, the Symmetrix array must be configured for device masking. The following requirements exist:

- ◆ There must be a valid VCMDB created on the Symmetrix array.
- ◆ The VCMDB must be mapped to all of the Fiber Adapters (FA) where the device masking is to take place.

- ◆ The VCM flag must be set on all of the FAs where the device masking is to take place.
- ◆ The Symmetrix device that is to be masked or unmasked must be mapped to all of the FAs where the device masking is to take place.

---

**Note:** SIU device masking does not perform any checks to determine if the Symmetrix device is currently accessible to other hosts on the SAN. It is your responsibility to prevent multiple hosts from using the same device.

---

If connected to a Symmetrix running Enginuity 5874 and higher, to perform Device Masking/Unmasking, at least one masking view must be present. Refer to [Chapter 3, “Masking Devices with Auto-provisioning Groups,”](#) for details about creating and using masking views.

Device masking is very useful when attempting to break open handles to a current volume. By masking the device from the host, Windows will be forced to completely unload the I/O stack for that device, thus removing all open handles. The device can then be unmasked and it will appear on the host without any open handles.

---

**Note:** In Windows 2008 devices must be properly uninstalled before being masked from the system. SIU performs the necessary uninstall process, however, the process can fail if there are open handles to a volume.

---

If you experience trouble masking devices on Windows 2008 use the `-force` flag to break any open handles. It is best to manually close any programs with open handles to the volume first, but specifying `-force` will break the handles automatically.

---

Using device masking to break the open handles to a volume may be necessary when using TimeFinder/Mirror in particular environments. Currently, it is recommended that the BCV be unmounted by SIU before establishing it. In most cases, this is sufficient to prevent corrupting the volume’s cache during the TimeFinder/Mirror establish. However, if the host is running monitoring agents or anti-virus software that maintains open handles to the BCV, the unmount will not be enough. Instead, you must mask the BCV prior to the TimeFinder/Mirror establish in order to remove the open handles. After the split operation, the BCV can be unmasked so that it is accessible to the host. This procedure is explained in [“For basic disks with open handles” on page 481.](#)

## Using volume flags

SIU provides three flag options that can be set on volumes, Symmetrix devices (and partitions), physical devices (and partitions), drive letters, and volume GUIDs using the `symmctl flag -set Flag` command.

Where *Flag* can be:

- ◆ READONLY — Sets the volume as read only.
- ◆ HIDDEN — Sets the volume as hidden. Volumes flagged as HIDDEN cannot be identified by `-guid` for any SIU command.
- ◆ NO\_DEFAULT\_DRIVE\_LETTER — Sets the volume to have no default drive letter.

For example, to set the HIDDEN flag on partition 4 of Symmetrix device 01e3, in Symmetrix array 309, enter:

```
symntctl flag -set hidden -sid 309 -symdev 01e3 -part 4
```

To clear an existing flag, use the `symntctl flag -clear` option, as follows:

```
symntctl flag -clear hidden -sid 309 -symdev 01e3 -part 4
```

Use the `symntctl flag -clear` command with the `-all` option to clear the flags for all disks or volumes, as shown:

```
symntctl flag -clear hidden -all
```

Refer to the *EMC Solutions Enabler Symmetrix CLI Command Reference* for the complete syntax of the `symntctl` command.

## Performing typical storage operations

The following are typical storage operations that use both SIU and TimeFinder/Mirror commands to manage the storage environment. The `sleep` and `timeout` utilities can be used interchangeably to cause a delay in the execution of scripts.

### Synchronizing the BCV

Using the `symntctl` and the `symmir` commands, the following procedures for basic and dynamic disks work with `devgroup1` to synchronize the BCV device (drive *x*) with the standard on primary and secondary servers. For more information on the `symmir` command, refer to *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*.

### For Windows Basic or Dynamic Disks

1. Unmount the volume on the secondary system:  

```
symntctl umount -drive x:
```
2. Establish the standard and BCV pair on the primary system:  

```
symmir -g devgroup1 est
```
3. Verify that the device is synchronized on the primary system:  

```
symmir -g devgroup1 verify -synched
```
4. Perform a flush on the primary system:  

```
symntctl flush -drive x:
```
5. Split the device on the primary system:  

```
symmir -g devgroup1 split
```
6. Perform a rescan on the secondary system:  

```
symntctl rescan
```
7. Mount the volume on the secondary system:  

```
symntctl mount -drive x: -vol volumex
```

### For Veritas Dynamic Disks

1. Unmount the volume on the secondary system:  

```
symntctl umount -drive x:
```



2. Deport the dynamic disk group on the secondary system and rescan:

```
vxdbg deport -g devgroup1  
symntctl rescan
```

3. Perform an establish on the primary system:

```
symmir -g devgroup1 est
```

4. Rescan the storage bus on the backup server to assure that the disk has been dropped and will not be accessed by Windows Plug and Play:

```
symntctl rescan
```

5. Verify that the device is synchronized on the primary system:

```
symmir -g devgroup1 verify -synched
```

6. Perform a flush on the primary system:

```
symntctl flush -drive x:
```

7. Split the device on the primary system:

```
symmir -g devgroup1 split
```

8. Perform a rescan on the secondary system:

```
symntctl rescan
```

9. Perform an import of the disk group on the secondary system and rescan:

```
vxdbg import -g devgroup1 -C  
symntctl rescan
```

10. Mount the volume on the secondary system:

```
symntctl mount -drive x: -vol volumex
```

## For basic disks with open handles

To ensure that cache corruption does not occur when a BCV has open handles, use the following procedure:

---

**Note:** This procedure should be used for all TimeFinder operations if the BCV volume has open handles (usually due to a monitoring agent or anti-virus software).

---

1. Unmount the volume on the secondary system:

```
symntctl umount -drive x: -force
```

2. Establish the standard and BCV pair on the primary system:

```
symmir -g devgroup1 est
```

3. Verify that the device is synchronized on the primary system:

```
symmir -g devgroup1 verify -synched
```

4. Perform a flush on the primary system:

```
symntctl flush -drive x:
```

5. Split the device on the primary system:

```
symmir -g devgroup1 split
```

6. Mount the volume on the secondary system:

```
symntctl mount -drive x: -vol volumex
```

## Restoring Veritas dynamic disks with TimeFinder

The following procedure restores a BCV of a dynamic disk that uses both SIU and TimeFinder/Mirror commands:

1. Unmount any drive letters or repase points currently assigned to the BCVs in the dynamic disk group on the backup server:

```
symntctl umount -drive x:
```

2. Deport the dynamic disk group on the backup server and rescan:

```
vxdg deport -g DgName  
symntctl rescan
```

3. Unmount any drive letters or repase points currently assigned to the standard devices in the dynamic disk group on the production server:

```
symntctl umount -drive x:
```

This step is necessary to ensure that the standard device is not written to during the restore (since all writes are committed to the BCV). Do not use `mountvol` to perform this dismount.

4. Deport the dynamic disk group on the production server and rescan:

```
vxdg deport -g DgName  
symntctl rescan
```

### CAUTION

**Failure to perform this step may leave the dynamic disk group in an unrecoverable state on the production server.**

5. Restore the TimeFinder/Mirror device group:

```
symmir -g DgName restore
```

6. Rescan the storage bus on the backup server:

```
symntctl rescan
```

The dynamic disk group and all of its dynamic disks are listed as unreadable, and are therefore, offline.

7. Split the TimeFinder/Mirror device group with the following commands:

```
symmir -g DgName verify -synched  
symmir -g DgName split
```

8. Rescan the storage bus on the backup server with the following command:

```
symntctl rescan
```

The dynamic disk group and all of its dynamic disks are readable, but are still offline and foreign.

9. Rescan the storage bus on the production server:

```
symntctl rescan
```

The dynamic disk group and all of its dynamic disks are present, but offline and foreign on the production server.

10. Import the dynamic disk group on the backup server:

```
vxdg import -g DgName -C
```

11. Rescan the storage bus on the backup server:

```
symntctl rescan
```

12. Import the dynamic disk group on the production server:

```
vxdg import -g DgName -C
```

The dynamic disk group and all of its dynamic disks are online and healthy.

13. Rescan the storage bus on the production server:

```
symntctl rescan
```

The dynamic disk group and all of its dynamic disks are present, but offline and foreign on the production server.

14. Remount each of the volumes in the disk group on the production server:

```
symntctl mount -drive x: -vol volumex
```

Use the LDM snap-in, SIU CLI, for this mount.

## Creating a backup using TimeFinder/Snap

This example starts with the VDEVS unmasked (available) to the backup/reporting host. Automount should be disabled. Disabled is the default and can be checked through the *diskpart* CLI.

1. Create the snap devices:

```
symnsnap -g test create -nop
```

2. Activate the snap devices:

```
symnsnap -g test activate -nop
```

3. Plug and Play rescan to discover the VDEVs in their Ready state:

```
symntctl rescan  
timeout 30
```

4. Update the SYMAPI database for SIU mount operations:

```
symcfg dis
```

5. The VDEVs should now be in their Ready state. Force Windows to rescan the device layout of the VDEVs:

```
symntctl update -sid 54 -symdev 3C0
```

6. Mount the snap devices to desired mount points:

```
symntctl mount -drive w: -symdev 3C0 -sid 54
```

7. When the backup or reporting is finished, unmount the snap devices:

```
symntctl umount -drive w:
```

8. Terminate the snap devices:

```
symsnap -g test terminate -nop
```

9. The VDEVs should now be in their terminated state. Force Windows to rescan the device layout of the VDEVs:

```
symntctl update -sid 54 -symdev 3C0
```

## Creating a backup using TimeFinder/Clone

This example starts with the BCVs unmasked (available) to the backup/reporting host. Automount should be disabled. Disabled is the default and can be checked through the "diskpart" CLI. Any volumes on the BCVs should have been previously unmounted by SIU.

1. Create or recreate the clone session using one or the other of these commands:

```
symclone create -g test  
symclone recreate -g test
```

2. Activate the clone devices:

```
symclone activate -g test
```

3. Mount the clone devices:

```
symntctl mount -drive w: -sid 54 -symdev 456
```

4. When backup or reporting is finished, unmount the clone devices:

```
symntctl umount -drive w
```

## SIU examples

This section provides examples using the `symntctl` command.

To flush any unwritten file system buffers to drive E, enter:

```
symntctl flush -drive E:
```

To flush any unwritten file system buffers to mount point E:\drivex, enter:

```
symntctl flush -path E:\drivex
```

To flush any unwritten file system buffers to volume volumex, enter:

```
symntctl flush -vol volumex
```

To flush a volume with a GUID of Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}, enter:

```
symntctl flush  
-guid Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}
```

To show the physical configuration for disk 3, enter:

```
symntctl show -pd disk3
```

To show the following example output enter:

```
symntctl show -vol HarddiskVolume7
```

To mount a basic volume called volumex to drive x, enter:

```
symntctl mount -drive x -vol volumex
```

To mount a dynamic volume called volumex of volumegroupx to mount point x:\volumex, enter:

```
symntctl mount -path x:\volumex -vol volumex  
-g volumegroupx
```

To mount a volume with the a GUID of Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303} to drive x, enter:

```
symntctl mount -drive x  
-guid Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}
```

To unmount a volume associated with drive x, enter:

```
symntctl umount -drive x
```

To unmount a volume called volumex of volumegroupx, enter:

```
symntctl umount -vol volumex -g volumegroupx
```

To unmount a volume with a GUID of Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303} from drive x, enter:

```
symntctl umount  
-guid Volume{ecba40b5-d522-11d5-98cb-00b0d0b03303}
```

To change the signature of disk 4 to 1ED67A70, enter:

```
symntctl signature -pd disk4 -sig 1ED67A70
```

To erase the signature of disk 4, enter:

```
symmtctl signature -pd disk4 -erase
```

To initialize the signature of all disks that have 0 for their signatures, enter:

```
symmtctl signature -initialize
```

To remove a Symmetrix device from the host, enter:

```
symmtctl mask -pd disk4
```

or

```
symmtctl mask -sid 000190300186 -symdev 3ED
```

To present a Symmetrix device to the host, enter:

```
symmtctl unmask -sid 000190300186 -symdev 3ED
```

## XML examples

SIU provides the ability to specify the output format for the `symmtctl list` and `symmtctl show` commands. Here are examples of XML output.

```
symmtctl.exe show -vol HarddiskVolume2 -output XML_ATTRIBUTE
```

```
<?xml version="1.0" standalone="yes" ?>
<SymCLI_ML>
  <Volume name="HarddiskVolume2" guid="e79fbcba-cdae-11dc-a419-005056b841fa" flag="PERMANENTLY
DISMOUNTED">
    <Extent name="\\.\PhysicalDrive3" starting="32256" length="526385664" symid="000190300186"
symdev="3DC" />
  </Volume>
</SymCLI_ML>
```

```
symmtctl.exe show -pd \\.\PhysicalDrive3 -output XML_ELEMENT
```

```
<?xml version="1.0" standalone="yes" ?>
<SymCLI_ML>
  <Disk>
    <name>\\.\PhysicalDrive3</name>
    <disktype>basic</disktype>
    <partitiontype>MBR</partitiontype>
    <size>4314</size>
    <symid>000190300186</symid>
    <symdev>3DC</symdev>
    <signature>5C02D0EB</signature>
    <status>online</status>
    <diskaddress>Port1Path0Target3Lun0</diskaddress>

<devicepath>\\?\scsi#disk&ven_emc&prod_symmetrix&rev_5771#4&3a739529&0&030#{53f56307-b6bf-11d
0-94f2-00a0c91efb8b}</devicepath>
    <adapter>LSI Logic PCI-X Ultra320 SCSI Host Adapter</adapter>
    <Volume>
      <name>HarddiskVolume2</name>
      <starting>32256</starting>
      <length>526385664</length>
    </Volume>
  </Disk>
</SymCLI_ML>
```

# APPENDIX G

## Managing Legacy Time Windows

This appendix describes the procedure used in Enginuity versions prior to 5875 for managing time windows in Optimizer, FAST and FAST VP. The new time windows procedure, using the `symtw` command, is explained in [“Setting time windows” on page 296](#).

- ◆ [Setting time windows.....](#) 488

## Setting time windows

**Note:** Time windows can be set for Optimizer or FAST. Both tools use the same command file format for time windows.

To set the swap time or performance time windows, use the `symoptmz` command with the following syntax in the command file:

```
set time_window id=tw_id,
    type=<SWAP | PERF>,
    flag=<INCLUDE | EXCLUDE>,
    period=<ONCE | WEEKLY | WEEKLY_BY_DAY>,
    starting=DateTime,
    ending=DateTime,
    [days=DayList,
    start_time=hh:mm,
    end_time=hh:mm];
```

Table 31 describes each parameter.

**Table 31** Optimizer time window parameters

Parameter	Description
time_window	Specifies a user-defined name that identifies this time window.
type	Defines the kind of time window being defined. Possible values are: SWAP: Defines whether a swap is (INCLUDE) or is not (EXCLUDE) performed in the given time window, according the flag setting. PERF: Defines whether the array's performance is monitored (INCLUDE) or not (EXCLUDE) in the given time window, according the flag setting. Using this setting, you could, for example, configure Optimizer to monitor on weekdays from 9 - 5 and only allow swaps on Sundays.
flag	Defines the option for the time window type. Possible values are: INCLUDE — Perform the action during this time. EXCLUDE — Do not perform the action during this time.
period	Defines the frequency of occurrence. Possible values are: ONCE means the action will occur only one time. WEEKLY_BY_DAY requires the individual days of the week to be specified. The time window applies to each day specified and each day has its own time window. WEEKLY setting is equivalent to a <i>range</i> . The <code>starting</code> parameter identifies the first of a series of consecutive days that the recurring time period applies to, and the next day in the list serves as the ending day of the week to which the time period applies.
DateTime	Identifies the date and time in the form of <code>MMDDYYYY:HHMMSS</code> .
DayList	Specifies a comma-separated combination of MON, TUE, WED, THU, FRI, SAT, or SUN. For the case of WEEKLY, <i>DayList</i> must begin with one of the following: MON_START, TUE_START, WED_START, THU_START, FRI_START, SAT_START, or SUN_START, which identifies the first of a series of consecutive days to which the time window applies. The next entry identifies the day of the week, which concludes the range of days.
start/end_time	Defines the start time <i>hh:mm</i> and end time as <i>hh:mm</i> .

## Clearing a time window

To clear a time window, use the following form:

```
clear time_window type=<SWAP | PERF>;
```





