

# An Overview of Groups in EMC Symmetrix and Solutions Enabler Environments

*A Detailed Review*

---

## **Abstract**

This white paper is an overview of the various types of groups that can be created in EMC<sup>®</sup> Symmetrix<sup>®</sup> and Solutions Enabler environments. It covers how these groups are created and how they are used to monitor and control Symmetrix storage arrays.

May 2010

---

---

Copyright © 2006, 2008, 2009, 2010 EMC Corporation. All rights reserved.

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

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS.” EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com

All other trademarks used herein are the property of their respective owners.

Part Number h2313.4

---

## Table of Contents

<b>Executive summary .....</b>	<b>4</b>
<b>Introduction .....</b>	<b>4</b>
Audience .....	4
<b>Static SRDF groups .....</b>	<b>5</b>
<b>Dynamic SRDF groups .....</b>	<b>6</b>
<b>SRDF group numbers .....</b>	<b>7</b>
<b>SRDF groups and concurrent SRDF .....</b>	<b>7</b>
<b>Device groups .....</b>	<b>9</b>
<b>Disk groups .....</b>	<b>10</b>
<b>Composite groups .....</b>	<b>12</b>
<b>SRDF consistency groups .....</b>	<b>13</b>
<b>Group Name Services.....</b>	<b>14</b>
<b>Auto-provisioning Groups .....</b>	<b>15</b>
<b>Interaction between group types.....</b>	<b>16</b>
<b>Typical use of different group types .....</b>	<b>17</b>
<b>Conclusion .....</b>	<b>18</b>
<b>References .....</b>	<b>18</b>

---

## Executive summary

In an EMC® Solutions Enabler environment several types of groups are commonly used to monitor and control Symmetrix® storage arrays such as Symmetrix DMX™ and now the VMAX™ Series arrays. It is important to understand how these groups are configured and implemented and how they interact so that users can most effectively and efficiently manage their storage environment. This knowledge is becoming even more critical as these environments become larger and more complex. This white paper provides an introduction to the types of groups that can be created and examples of how they are most commonly used.

## Introduction

This white paper describes the uses for, and the differences between, the various types of groups that are used in Symmetrix and Solutions Enabler environments. Grouping devices serves a variety of purposes including:

- To configure and manage remote replication (static and Dynamic SRDF® groups)
- To manage groups of devices from a single local Symmetrix array (device groups)
- To manage groups of devices spread across multiple local Symmetrix arrays (composite groups)
- To ensure remote data consistency (RDF consistency groups)
- To manage device and composite groups that must be available to multiple hosts that are attached to a Symmetrix (Global Name Services)
- To manage groups of host initiators, front-end ports, and storage volumes in order to simplify the mapping and masking process (Auto-provisioning Groups)

Static and Dynamic SRDF groups are the method used in a Symmetrix environment to map devices to remote adapters for SRDF replication. Understanding the properties of both group types will allow storage administrators to more effectively manage their SRDF environment.

Symmetrix device groups have been the primary method that storage administrators have used to manage their devices for both local and remote replication since Solutions Enabler was introduced. As storage environments have continued to grow, the management of increasingly larger numbers of devices contained in multiple local and remote Symmetrix arrays has become more of a challenge. This has been further complicated by large applications that use devices spread across multiple arrays and the requirement that consistent copies of data be available both locally and remotely.

This growth has led to the development of composite device groups, consistency groups, and Global Name Services. All of these developments add functionality and provide administrators with more flexibility in managing and controlling their storage environment.

Besides explaining the different group types that can be used to manipulate Symmetrix devices, this paper also discusses the following:

- How the different group types interact with each other
- How each is typically used in production environments

---

**Note:** In some of the command output, unnecessary legend information has been removed for the sake of brevity. In other output, an ellipsis (...) indicates that unnecessary output has also been removed.

---

## Audience

This paper is intended for the storage administrator or manager who is new to Symmetrix disk arrays and Solutions Enabler software, and is looking for an introduction to the various group types used in this environment.

## Static SRDF groups

SRDF groups are used to define relationships between two Symmetrix storage arrays that are logically connected with EMC Symmetrix Remote Data Facility (SRDF). SRDF groups define the relationships between the following:

- SRDF director ports in a local array
- Corresponding SRDF ports in a remote array

Each Symmetrix volume that is configured as an SRDF device must be assigned to an SRDF group. However, the source (R1) volume and the target (R2) volume are not required to belong to groups with the same group number. For example, an R1 in a local Symmetrix can be in SRDF group 01 and can be paired with an R2 in SRDF group 0A in a remote Symmetrix. The addition of SRDF devices to an SRDF group is the mechanism that is used to assign SRDF devices to a particular RA port or ports.

SRDF groups that are explicitly configured in the Symmetrix bin file are named static SRDF groups. Static SRDF groups are created and deleted by using a bin file change. At Enginuity™ 5x68 and earlier, up to 16 static SRDF groups can be configured. Beginning with 5669 code, up to 64 static SRDF groups can be configured per Symmetrix array.

The number of SRDF groups that can be configured per SRDF director port is dependent upon the type of director in use. There are currently three types of directors available for SRDF connectivity:

- ESCON remote adapter (RA)
- Fibre Channel remote adapter (RF)
- GigE remote adapter (RE) – Symmetrix 8000 series or later
- This includes Multiprotocol Channel Director (MPCD) for Symmetrix DMX™ or later, which can be configured for SRDF over GigE

With ESCON RAs, only one SRDF group can be configured per remote director port. With other available SRDF director types, up to 16 SRDF groups can be configured per remote director port but the recommended maximum is six.

Symmetrix SRDF groups can be displayed by using the **symcfg list -rdfg** command:

```
# symcfg list -rdfg ALL -sid 0731
```

```
Symmetrix ID : 000185500731
```

```

          S Y M M E T R I X   R D F   G R O U P S

Local          Remote          Group          RDFA Info
-----
LL
RA-Grp (sec)  RA-Grp  SymmID      T  Name      Flags  Dir  Flags  Cycle  Pri
-----
1 ( 0)      0    1 ( 0) 000185500864  S   .X.      F-H   ---   -     -

```

Legend:

```

Group (T)ype      : S = Static, D = Dynamic
Group Flags      :
  Prevent Auto (L)ink Recovery      : X = Enabled, . = Disabled
  Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
  Link (D)omino      : X = Enabled, . = Disabled
Director (C)onfig : F-S = Fibre-Switched, F-H = Fibre-Hub
                  G = GIGE, E = ESCON, T = T3, - = N/A
RDFA Flags      :
(C)onsistency    : X = Enabled, . = Disabled, - = N/A
(S)tatus        : A = Active, I = Inactive, - = N/A
(R)DFA Mode     : S = Single-session, M = MSC, - = N/A

```

---

## Dynamic SRDF groups

Starting at 5669 code, users can create SRDF groups and assign them to Fibre Channel directors (in a switched SRDF environment only) by using SRDF host-based control software. This functionality is also provided for GigE SRDF. If dynamic SRDF devices have been created in the Symmetrix, users can add them to dynamic SRDF groups and configure their own SRDF environment without the need for bin file changes to be made by EMC Customer Support.

SRDF groups can also be deleted dynamically. Dynamic SRDF group information is not written to the Symmetrix bin file. However, changes made to the dynamic SRDF configuration are persistent through a Symmetrix power cycle and IMPL.

Dynamic groups also allow SRDF configuration changes that can alleviate performance issues. For example, to reduce the amount of devices that are assigned to a busy RF port, a dynamic SRDF group may be moved to another properly configured RF by using the `symrdf modifygrp --add` and `symrdf modifygrp --remove` commands.

---

**Note:** If devices need to be moved individually from one SRDF group to another, they must be converted to non-SRDF devices; the RDF pairs can then be re-created and placed into the desired group. This action requires a full establish.

---

The listing of SRDF group types can be restricted by using the `--static` or `--dynamic` options. If no groups of the specified type exist, the command prompt returns without writing anything to the console:

```
# symcfg list -rdfg all -dynamic -sid 80
```

```
Symmetrix ID : 000187990080
```

```
          S Y M M E T R I X   R D F   G R O U P S
-----
Local          Remote          Group          RDFA Info
-----
RA-Grp  LL          RA-Grp  SymmID          T   Name          Flags Dir  Flags Cycle  Pri
-----
3 ( 2)  10          3 ( 2)  000187490028   D  test          XX.. F-S  -IS-   30   33
4 ( 3)  10          4 ( 3)  000190100594   D  test594        XX.. F-S  -IS-   30   33
```

Legend:

```
? : Unknown
Group (T)ype : S = Static, D = Dynamic
Director (C)onfig : F-S = Fibre-Switched, F-H = Fibre-Hub
                  G = GIGE, E = ESCON, T = T3, - = N/A
Group Flags :
  Prevent Auto (L)ink Recovery : X = Enabled, . = Disabled
  Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
  Link (D)omino : X = Enabled, . = Disabled
  (S)TAR mode : N = Normal, R = Recovery, . = OFF
RDF A 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
```

---

**Note:** In the initial release of SRDF/A, asynchronous capable devices and device groups could not be configured by using Dynamic SRDF. This configuration capability became available with the release of 5671 code.

---

---

## SRDF group numbers

In EMC documentation, Symmetrix SRDF groups may also be referred to as RA groups or RDF groups. These terms essentially mean the same thing, though they are numbered differently by Solutions Enabler and Symmetrix Enginuity.

For example, the following is taken from the output of the command `symdev show 0272 -sid 22`:

```
RDF Information
{
  Device Symmetrix Name      : 0272
  RDF Type                   : R1
  RDF (RA) Group Number    : 36                (23)

  Remote Device Symmetrix Name : 0272
  Remote Symmetrix ID         : 000187400135
```

The output shows both an RDF and an RA group number for R1 device 0272. The RDF group number for the device, which is noted in decimal, is 36. This decimal value is the number that must be used with Solutions Enabler commands that require that an RDF group be specified with the `-RDFG` option.

The RA group number is the identifier for the device in Symmetrix Enginuity and is notated in hex. In this case, the group number is 0x23. This number is used by EMC Customer Support personnel only when making bin file changes using EMC internal configuration tools; it should never be needed by the user.

---

**Note:** 0x23 is 35 in decimal. The reason for the discrepancy between the numbers is due to the fact that decimal RDF group numbering begins at 01, but hex numbering begins at 00.

---

## SRDF groups and concurrent SRDF

With concurrent SRDF, a source (R1) device has two associated target (R2) devices, each in a different remote Symmetrix array. In this environment a user has the option of performing some SRDF operations, such as a `symrdf split`, against both remote devices concurrently or against a specific R1/R2 pair. There are also some operations, such as a `symrdf restore`, that may be performed only on specific R1/R2 pairs. Solutions Enabler and Symmetrix Enginuity use SRDF group numbers to distinguish which R1/R2 pairs to act on. For this to occur, each of the concurrent SRDF target devices must belong to a different SRDF group.

In the following example, local R1 devices 0272 and 0273 each have two target devices, one in SRDF group number 36 and the other in group 54.

The `symrdf` commands can be run by specifying individual SRDF groups:

```
# symrdf -g concurrent_dg query -rdfg 36

Device Group (DG) Name      : concurrent_dg
DG's Type                   : RDF1
DG's Symmetrix ID          : 000187800022
Remote Symmetrix ID        : 000187400135
RDF (RA) Group Number      : 36 (23)
```

Source (R1) View					Target (R2) View					MODES	
Standard	ST				LI	ST					
	A				N	A					
Logical	T	R1 Inv	R2 Inv		K	T	R1 Inv	R2 Inv		RDF Pair	
Device	Dev	E	Tracks	Tracks	S	Dev	E	Tracks	Tracks	MDA	STATE
DEV001	0272	RW	0	0	RW	0272	NR	0	0	S..	Synchronized
DEV002	0273	RW	0	0	RW	0273	NR	0	0	S..	Synchronized
Total											
	Track(s)		0	0				0	0		
	MB(s)		0.0	0.0				0.0	0.0		

```
# symrdf -g concurrent_dg query -rdfg 54
```

```
Device Group (DG) Name      : concurrent_dg
DG's Type                   : RDF1
DG's Symmetrix ID          : 000187800022
Remote Symmetrix ID        : 000187400732
RDF (RA) Group Number      : 54 (35)
```

Source (R1) View					Target (R2) View					MODES	
Standard	ST				LI	ST					
	A				N	A					
Logical	T	R1 Inv	R2 Inv		K	T	R1 Inv	R2 Inv		RDF Pair	
Device	Dev	E	Tracks	Tracks	S	Dev	E	Tracks	Tracks	MDA	STATE
DEV001	0272	RW	0	0	RW	0202	WD	0	0	S..	Synchronized
DEV002	0273	RW	0	0	RW	0203	WD	0	0	S..	Synchronized
Total											
	Track(s)		0	0				0	0		
	MB(s)		0.0	0.0				0.0	0.0		

They can also be run by specifying all SRDF groups:

```
# symrdf -g concurrent_dg query -rdfg ALL
```

```
Device Group (DG) Name      : concurrent_dg
DG's Type                   : RDF1
DG's Symmetrix ID          : 000187800022
Remote Symmetrix ID        : 000187400135
RDF (RA) Group Number      : 36 (23)
Remote Symmetrix ID        : 000187400732
RDF (RA) Group Number      : 54 (35)
```

Source (R1) View					Target (R2) View					MODES	
Standard	ST				LI	ST					
	A				N	A					
Logical	T	R1 Inv	R2 Inv		K	T	R1 Inv	R2 Inv		RDF Pair	
Device	Dev	E	Tracks	Tracks	S	Dev	E	Tracks	Tracks	MDA	STATE
DEV001	0272	RW	0	0	RW	0272	NR	0	0	S..	Synchronized
		RW	0	0	RW	0202	WD	0	0	S..	Synchronized
DEV002	0273	RW	0	0	RW	0273	NR	0	0	S..	Synchronized
		RW	0	0	RW	0203	WD	0	0	S..	Synchronized
Total											
	Track(s)		0	0				0	0		
	MB(s)		0.0	0.0				0.0	0.0		

---

## Device groups

Symmetrix device groups are user-defined objects that are used to view and manage groups of Symmetrix devices. Solutions Enabler commands can be run against these groups of devices as a whole or against individual devices within the group.

Device group definitions are contained in the SYMAPI configuration database file, called `symapi_db.bin`, which is located in the following directories by default:

- UNIX        `/var/symapi/db`
- Windows    `C:\Program Files\EMC\Symapi\db`

Prior to Solutions Enabler 6.3, a device could only belong to one device group that is configured in a single `symapi_db.bin` file. In other words, device 001 could belong to a device group named `oradg` on host A, but it could not belong to a group named `ora2dg` that is also created on host A. It may, however, belong to any group on host B. Starting in Solutions Enabler 6.3, the `SYMAPI_ALLOW_DEV_IN_MULT_GRP` parameter can be set to `ENABLE` in the SYMAPI options file to allow devices to belong to multiple device groups in the same `symapi_db.bin` file.

Device group names are assigned at the time that the group is created but can be modified later. Device group names must be unique to the host that is running the Solutions Enabler software and can contain up to 31 characters.

Each device group has a type that indicates the device configuration of the standard devices that it will contain. The three possible types are `REGULAR`, `RDF1`, or `RDF2`.

- `REGULAR` device groups can contain only standard devices that are not configured for SRDF remote replication.
- `RDF1` groups can contain only SRDF source (or R1) standards.
- `RDF2` groups can contain only SRDF target (or R2) standards.

The type of the group is set when the group is created and must contain only devices that match the group type. If no group type is specified, the group type will be `REGULAR`:

```
# symdbg create testdg
# symdbg list
```

```
                D E V I C E      G R O U P S

Name            Type      Valid  Symmetrix ID  Devs  Number of
                GKs   BCVs  VDEVs
testdg          REGULAR  N/A   N/A           0     0     0     0
```

Group types are specified by using the `-type` argument with the `symdbg create` command:

```
# symdbg create r1_testdg -type RDF1
# symdbg create r2_testdg -type RDF2
# symdbg list
```

```
                D E V I C E      G R O U P S

Name            Type      Valid  Symmetrix ID  Devs  Number of
                GKs   BCVs  VDEVs
testdg          REGULAR  N/A   N/A           0     0     0     0
r1_testdg       RDF1     N/A   N/A           0     0     0     0
r2_testdg       RDF2     N/A   N/A           0     0     0     0
```

---

The device group type is used only to determine what type of standard devices can be added to the group. If BCVs are added, they can be either Regular or RDF devices regardless of the group type.

Symmetrix device groups can contain devices from a single local Symmetrix array and from a single RDF (RA) group. Attempting to add devices from more than one local Symmetrix array will cause the command to fail:

```
# symld -g testdg add dev 196 -sid 31
# symld -g testdg add dev 015 -sid 42
```

Devices cannot be added; the existing group has devices from a different Symmetrix

Solutions Enabler operations are run against Symmetrix device groups by using the **-g** option. For example, the command to run an incremental EMC TimeFinder® establish against all standard and BCV pairs within a device group named “testdg” would be the following:

```
symmir -g testdg establish
```

---

**Note:** Symmetrix device groups that are configured as RDF1 or RDF2 groups should not be confused with SRDF groups. SRDF groups exist in the Symmetrix bin file or in Symmetrix cache. Symmetrix device groups exist in the symapi\_db.bin file.

---

## Disk groups

A disk group is a group of physical disks that have been created in the Symmetrix bin file. Disk groups are formed when the bin file is initially written or when additional physical disks are added to the system. They are used to separate physical disks by disk type. For example, if a Symmetrix array has both 146 GB 10,000 rpm drives and 500 GB 7,500 rpm drives, the two drive types can be separated into two different disk groups. This allows users to create multiple performance tiers within a single Symmetrix frame. This is accomplished by specifying a disk group when Symmetrix logical volumes are created so that the volumes are confined to a particular size and speed of disk.

Disk groups can be viewed using Solutions Enabler. For example, the **syndisk** command, which is used to view the configuration and status of disks and their hypervolumes, contains both a **-by\_diskgroup** and a **-disk\_group** option.

When the **-by\_diskgroup** option is used with **syndisk list**, the output displays the disks listed by disk group number starting with group 0:

```
# syndisk list -by_diskgroup -sid 32
```

```
Symmetrix ID          : 000190103232
Disks Selected        : 240
```

Ident	Symb	Int	TID	Vendor	Type	Hypers	Capacity(MB)		
							Total	Free	Actual
Disk Group 0:									
DF-1A	01A	C	0	SEAGATE	C146LDF	31	140014	103331	140014
DF-1A	01A	C	2	SEAGATE	C146LDF	34	140014	107272	140014
DF-1A	01A	C	4	SEAGATE	C146LDF	27	140014	104260	140014
DF-1A	01A	C	6	SEAGATE	C146LDF	20	140014	109776	140014
DF-1A	01A	C	8	SEAGATE	C146LDF	16	140014	64205	140014

...

Disk Group 1:

DF-2A	02A	C	1	SEAGATE	C300LPF	65	286102	131058	286102
-------	-----	---	---	---------	---------	----	--------	--------	--------

```
DF-2A 02A C 3 SEAGATE C300LPF 67 286102 128453 286102
DF-2A 02A C 5 SEAGATE C300LPF 61 286102 127582 286102
DF-2A 02A C 7 SEAGATE C300LPF 0 0 0 286102
DF-2A 02A D 0 SEAGATE C300LPF 67 286102 133332 286102
```

...

When the **-disk\_group** option is used with **symdisk list** and a particular group is selected, the output displays the disks in the disk group beginning with the lowest numbered DA:

```
# symdisk list -disk_group 2 -sid 32
```

```
Symmetrix ID          : 000190103232
Disks Selected       : 240
```

Ident	Symb	Int	TID	Vendor	Type	Hypers	Capacity(MB)		
							Total	Free	Actual
DF-2A	02A	C	9	SEAGATE	BL500LP	80	476837	397840	476837
DF-2A	02A	C	B	SEAGATE	BL500LP	82	476837	395452	476837
DF-2A	02A	C	D	SEAGATE	BL500LP	81	476837	395125	476837
DF-2A	02A	D	8	SEAGATE	BL500LP	81	476837	396399	476837
DF-2A	02A	D	A	SEAGATE	BL500LP	82	476837	395452	476837

...

Symmetrix devices can be related to a disk group by using the **symdev list** command and specifying a disk group number:

```
# symdev list -disk_group 0 -sid 32
```

```
Symmetrix ID: 000190103232
```

Sym	Physical	Device Name	Directors		Config	Device	Attribute	Sts	Cap (MB)
			SA	:P DA :IT					
0043	Not Visible		07A:0	01C:C2	2-Way Mir	N/Grp'd	RW	6	
0044	Not Visible		07A:0	16C:C3	2-Way Mir	N/Grp'd	RW	6	
0045	Not Visible		07A:0	01B:D2	2-Way Mir	N/Grp'd	RW	6	
0046	Not Visible		07A:0	16B:D3	2-Way Mir	N/Grp'd	RW	6	
0047	Not Visible		07A:0	01D:D2	2-Way Mir	N/Grp'd	RW	6	

...

The physical disk and disk group that a volume belongs to can also be seen in the Back End Disk Director Information section of the **symdev show** output:

```
# symdev show 043 -sid 32
```

```
Device Physical Name      : Not Visible
Device Symmetrix Name     : 0043
Device Serial ID         : N/A
Symmetrix ID             : 000190103232
```

...

```
Back End Disk Director Information
{
  Hyper Type                : Data
  Hyper Status              : Ready                (RW)
  Disk [Director, Interface, TID] : [01C, C, 2]
  Disk Director Volume Number : 53 (0x34)
  Hyper Number              : 18
  Disk Capacity             : 140014m
  Disk Group Number        : 0

  Hyper Type                : Data
  Hyper Status              : Ready                (RW)
```

```

Disk [Director, Interface, TID]      : [01D, C, 3]
Disk Director Volume Number         : 53 (0x34)
Hyper Number                         : 18
Disk Capacity                       : 140014m
Disk Group Number                   : 0
}

```

Although users do not have the ability to create or modify disk groups, they can be used in certain Solutions Enabler operations. These include Symmetrix Optimizer migrations using Virtual LUN technology and device creation using the Configuration Change component of SYMCLI.

With Symmetrix Optimizer, a disk group can be specified as the target of a virtual LUN migration. This allows the user to migrate application data between performance tiers as business needs change. With the Configuration Change component, the user can specify a disk group when creating Symmetrix devices. This allows the user to control which group of physical disks Symmetrix devices will be created from.

## Composite groups

A composite group follows the same rules as a Symmetrix device group except it can contain devices that span multiple local Symmetrix arrays and multiple RDF (RA) groups. For example, with composite groups it is possible to make a consistent, restartable local copy of a database that is using volumes spanning two local Symmetrix arrays. Without composite groups it would not be possible to guarantee that all of the BCVs would be split at the same point in time.

Composite groups are created and managed by using the **symcgs** command:

```

# symcgs create testcg
# symcgs list

```

C O M P O S I T E    G R O U P S								
Name	Type	Valid	Number of		Number of			
			Syms	RAGs	Devs	BCVs	VDEVs	
testcg	REGULAR	Yes	0	0	0	0	0	

Unlike Symmetrix device groups, adding devices to a composite group is accomplished by using the **symcgs add dev** command. Note that instead of **-g**, composite group names are specified by using the **-cg** option.

```

# symcgs -cg testcg add dev 197 -sid 31
# symcgs -cg testcg add dev 016 -sid 42

```

The composite group now contains devices from two local Symmetrix arrays:

```

# symcgs list

```

C O M P O S I T E    G R O U P S								
Name	Type	Valid	Number of		Number of			
			Syms	RAGs	Devs	BCVs	VDEVs	
testcg	REGULAR	Yes	2	0	2	0	0	

Like Symmetrix device groups, composite groups are stored locally in the `symapi_db.bin` file.

---

## SRDF consistency groups

An SRDF consistency group is a composite group that is created with special properties to allow RDF devices in separate RDF groups to maintain dependent write consistency. The RDF groups within the consistency group can exist within a single Symmetrix array or they can be spread across multiple Symmetrix arrays. Write consistency is maintained by using either EMC PowerPath® or Enginuity Consistency Assist (ECA) for SRDF/Synchronous or Multi-Session Consistency (MSC) for SRDF/Asynchronous.

For example, consider an environment where a database is spread across two local Symmetrix arrays, each with its own target array at a remote DR site. If the SRDF link between one local Symmetrix and its target array is lost, remote replication would be halted for all R1 devices replicating between those arrays. Host processing would continue uninterrupted. This means that the R1 devices that exist on the other Symmetrix arrays and still have a valid link to the target Symmetrix would continue to accept host writes and propagate that data to the target Symmetrix. This would result in an inconsistent remote copy for the duration of both the link outage and the time required to resynchronize the devices after the link is restored.

ECA, PowerPath, and MSC preserve dependent write consistency at the target by continually monitoring the status of the SRDF devices and suspending all SRDF links if one or more R1 devices are unable to propagate data to their target R2 device. This guarantees a restartable remote copy of the database on the R2 devices. A consistent copy can be preserved for the duration of both the outage and the SRDF resynchronization by creating a local BCV or clone copy of the R2s before resynchronization is started.

Consistency groups must:

- Be RDF1 or RDF2 type composite groups.
- Contain standard devices.
- Be composed of devices under PowerPath control or registered with the RDF daemon.
- Be enabled for consistency.

When a composite group is created with the **symcg create** command, either **-ppath** for PowerPath consistency or **-rdf\_consistency** for MSC or ECA must be specified to make the group an RDF consistency group. Creating the composite group with the **-ppath** flag causes PowerPath to manage consistency protection for the group. Creating the composite group with the **-rdf\_consistency** flag causes the RDF daemon to manage the group's consistency protection.

In the following example, the devices in the group will be registered with PowerPath for consistency protection:

```
# symcg create r1_testcg -type RDF1 -ppath
# symcg -cg r1_testcg -sid 55 add dev 001
# symcg -cg r1_testcg -sid 55 add dev 002
# symcg list
```

```
          C O M P O S I T E   G R O U P S
Name          Type   Valid  Number of   Number of
              Type   Valid  Symms   RAGs   Devs   BCVs   VDEVs
r1_testcg     RDF1   Yes    1       1       2       0       0
```

---

Note: If both composite groups and consistency groups exist, the **-ppath** or **-rdf\_consistency** options can be used with **symcg list** to limit the type of groups that are displayed in the command output.

---

---

The RDF consistency group can now be enabled.

```
# symcg -cg r1_testcg enable
```

```
Execute a consistency 'Enable' operation for composite
group 'r1_testcg' (y/[n]) ? y
```

```
A consistency 'Enable' operation execution is
in progress for composite group 'r1_testcg'. Please wait...
```

```
The consistency 'Enable' operation successfully executed for
composite group 'r1_testcg'.
```

---

Note: Existing device groups can be converted to composite groups (with or without RDF consistency) by using the `symdg dg2cg` command.

---

## Group Name Services

Group Name Services (GNS) allows SYMAPI device group and composite group definitions to be stored in a repository on all Symmetrix arrays that are local to hosts that run Solutions Enabler. Any host with GNS enabled can view and use the device and composite groups that are contained in the repository regardless of what host actually ran the commands to create them. This provides redundancy in case of a failure that causes the host that normally runs Solutions Enabler to lose access to the Symmetrix. It also creates standards for device groups and device names that can be shared across all hosts in the environment.

The GNS repository is updated by a daemon named `storgnsd` that must be running on any host that performs SYMAPI management operations that use GNS.

To use GNS, the `SYMAPI_USE_GNS` entry in the SYMAPI options file must be set to `ENABLE`:

```
# pwd
/var/symapi/config

# cat options | grep SYMAPI_USE_GNS
# Parameter:                SYMAPI_USE_GNS
SYMAPI_USE_GNS                = ENABLE
```

Once the options file has been updated, the **storgnsd** daemon will start when Solutions Enabler attempts to connect to the GNS repository. It can also be started manually from the command line.

```
# stordaeomon start storgnsd
  Waiting for daemon to start.  This may take several seconds.
```

When enabled, all device group definitions are stored in the GNS repository instead of in the local SYMAPI database file.

Running `symdg show` against a device group indicates that the group is located in the GNS repository.

---

```
# symdbg show GNS_testdg
```

```
Group Name: GNS_testdg
```

```
Group Type                : REGULAR
Device Group in GNS       : Yes
Valid                     : N/A
Symmetrix ID              : N/A
Group Creation Time       : Wed Apr  5 13:53:36 2006
Vendor ID                 :
Application ID            :

Number of STD Devices in Group      : 0
Number of Associated GK's           : 0
Number of Locally-associated BCV's  : 0
Number of Locally-associated VDEV's : 0
Number of Remotely-associated VDEV's(STD RDF): 0
Number of Remotely-associated BCV's (STD RDF): 0
Number of Remotely-associated BCV's (BCV RDF): 0
Number of Remotely-assoc'd RBCV's (RBCV RDF) : 0
```

Existing device groups that resided in the SYMAPI database file now also reside in GNS:

```
# symdbg show testdrx
```

```
Group Name: testdrx
```

```
Group Type                : RDF1
Device Group in GNS       : Yes
Valid                     : Yes
Symmetrix ID              : 000187900546
Group Creation Time       : Tue Dec 21 13:23:27 2004
Vendor ID                 : EMC Corp
Application ID            : SYMCLI
```

```
...
```

Once GNS is enabled, the GNS daemon updates the repository when any changes are made to device, composite, or consistency groups. It also updates the PowerPath database and the RDF daemon with any changes so that remote consistency is maintained if consistency groups are configured.

---

**Note:** If composite groups are used, hosts must be locally attached to all Symmetrix arrays that contain devices in that composite group. If not, the composite group will be in an invalid state and will be unusable.

---

## Auto-provisioning Groups

Introduced in Solutions Enabler 7.0, the Auto-provisioning Groups feature provides an easier, faster way to provision storage, reducing labor and risk of error in Symmetrix VMAX arrays running Enginuity 5874. Most of the applications running on Symmetrix arrays require a fault-tolerant environment with clustered hosts as well as multiple paths to devices. The Auto-provisioning Groups feature was developed to make storage allocation easier and faster, especially in clustered and virtualized server environments.

Mapping and masking devices in previous versions of Solutions Enabler required a separate command for each initiator/port combination through which devices would be accessed. In Solutions Enabler 7.0, the `symaccess` command allows the user to create a group of devices (*storage group*), a group of director ports (*port group*), and a group of host initiators (*initiator group*), and associate them in a *masking view*. In simple configurations a masking view, along with an associated storage, port, and initiator group, can also be created for a single initiator on a single port with any number of devices using one command.

In either case, when the masking view is created, the devices are automatically masked. If the devices in the storage group are not mapped to the front-end ports in the port group, they will also be mapped before the operation completes. Once the `symaccess` command that creates the view completes, the devices are available to the initiators on the storage ports. No further Solutions Enabler operations are required.

After the masking view is created, any objects (devices, ports, or initiators) added to an existing group automatically become part of the associated masking view. This means that no additional steps are necessary to add additional devices, ports, or initiators to an existing configuration. All necessary operations to make them part of the configuration are handled automatically by Symmetrix Engenuity once the objects are added to the applicable group. This greatly reduces the number of commands required to add additional objects to an existing configuration and allows for easier storage allocation and de-allocation.

The following is an example of a masking view that contains an initiator group, port group, and storage group:

```
# symaccess show view prodhost -sid 54

Symmetrix ID          : 000192601254

Masking View Name     : prodhost
Initiator Group Name  : prodhost_hbas

  Host Initiators
  {
    WWN : 10000000c9767816
    WWN : 10000000c9767817
  }

Port Group Name       : prodhost_ports

  Director Identification
  {
    FA-7E:0
    FA-7F:1
  }

Storage Group Name    : prodhost_stor
```

Sym Dev	Dir:P	Physical Device Name	Host Lun	Attr	Cap(MB)
0833	07E:0	c3t5000097208139918d1s*	1		4314
	07F:1	c4t5000097208139959d1s*	1		4314
0834	07E:0	c3t5000097208139918d2s*	2		4314
	07F:1	c4t5000097208139959d2s*	2		4314
0835	07E:0	c3t5000097208139918d3s*	3		4314
	07F:1	c4t5000097208139959d3s*	3		4314
0836	07E:0	c3t5000097208139918d4s*	4		4314
	07F:1	c4t5000097208139959d4s*	4		4314
0837	07E:0	c3t5000097208139918d5s*	5		4314
	07F:1	c4t5000097208139959d5s*	5		4314
0838	07E:0	c3t5000097208139918d6s*	6		4314
	07F:1	c4t5000097208139959d6s*	6		4314

## Interaction between group types

There may be some or no relationship at all between the types of groups discussed in this paper. It depends on the type of group and the purpose that the group serves in the Symmetrix environment. For instance, in most situations, the devices in SRDF groups are not strictly tied to devices in device groups. In an environment where there is a single local Symmetrix replicating to a single remote Symmetrix, there may be a single SRDF group that contains all SRDF devices. The user may have four different applications in use, each with a set of devices that are being replicated over the RDF link. In this case, the user would likely have four Symmetrix device groups configured, one for each set of devices. In that example, there is little correlation between the SRDF group and the device groups. An exception to this would be with SRDF/A where SRDF groups and corresponding device groups should share the same devices.

---

Prior to the release of Engenuity 5671 only a single, static SRDF group could be configured with devices that were capable of running in asynchronous mode (SRDF/A). When SRDF/A was active, all SRDF/A capable devices were required to be running in asynchronous mode and all must be acted upon as a group when any Solutions Enabler commands were run against them. In other words, if you needed to split a device running in asynchronous mode, all devices in that SRDF group needed to be split in the same operation. Because of this requirement, in most implementations, all SRDF/A capable devices in a Symmetrix were put into the same Symmetrix device group. This way, they could be acted upon simultaneously by creating a 1:1 relationship between the SRDF group and device group.

Starting with the introduction of 5671 code, all SRDF groups are capable of running in asynchronous mode (SRDF/A). It should be noted, that if devices in a particular SRDF group are to run in asynchronous mode, that all devices must still run in asynchronous mode and all must be acted upon as a group. This means that the 1:1 relationship between SRDF groups and device groups still applies. In 5671 and later implementations, for example, 10 SRDF/A capable SRDF groups would normally require 10 Symmetrix device groups.

## Typical use of different group types

Unlike the other type of groups discussed in this paper, which are used as a means to monitor and manage Symmetrix devices, SRDF groups are required when SRDF devices are configured in the system. This is because SRDF groups are used to determine the local and remote RA port pairings in an SRDF environment. This holds true whether the group is static or dynamic. SRDF groups are not used for other purposes. However, the number of a particular SRDF group or groups may need to be referenced in Solutions Enabler commands as a means of specifying what SRDF pair the command should act on.

The other group types discussed in this paper are created to perform operations against groups of devices. They are normally configured so that devices that belong to specific hosts or applications are acted upon at the same time. The type of group that a user might want to implement is dependant upon the storage, host, and application environment.

Symmetrix device groups were introduced with the first release of Solutions Enabler and are the least complicated type of group to use and administer. They are generally created in simple configurations where both of the following apply:

- There are no applications that span multiple Symmetrix arrays or RA groups.
- Multiple hosts running Solutions Enabler will not be used to regularly manage the Symmetrix.

Composite groups were created to serve a simple but necessary function. Composite groups allow the user to create a group that contains devices from multiple Symmetrix arrays or RA groups. This is required when Solutions Enabler commands must be run simultaneously against devices that span multiple arrays.

RDF consistency groups were created to provide specific functionality within an SRDF environment. They are used only when a dependent write consistent copy of a database must be created on target devices in an SRDF/S or SRDF/A environment.

Group Name Services are used in environments where access to both device and composite groups across a set of Symmetrix arrays is desired. This is generally used in more complex environments where there is a need to be able to access device group information from multiple hosts. GNS eliminates the need to copy device or composite group information (when using **symdmg** or **symcrg export** and **import**) between Solutions Enabler hosts in case there is a connectivity issue that prevents one of the machines from accessing the Symmetrix. It also creates standards for device groups and device names that can be shared across all hosts in the environment.

In the past, it was typical for users to start with Symmetrix device groups and then move to other group types, if needed, as their storage environment grew and changed. Due to the increasing complexity of

---

storage environments today, it is becoming more common for customers to begin using Solutions Enabler with GNS enabled so the hosts from which the environment is monitored and managed can be easily changed.

## Conclusion

In a Symmetrix environment, various group types allow the user to monitor and control their storage arrays. It is important to understand how these groups are configured and implemented and how they interact so that users can most efficiently manage their environment. This is becoming even more critical as storage environments become larger and more complex. This paper is meant as an introduction to the types of groups that can be created and not as a comprehensive guide for implementation. For more details on the topics covered in this paper, please see the product guides referenced in the following section.

## References

For more information, please refer to the following documentation:

- *EMC Solutions Enabler Symmetrix Array Management CLI Version 7.0 Product Guide*
- *EMC Symmetrix Remote Data Facility (SRDF) Product Guide*
- *EMC Solutions Enabler Symmetrix SRDF Family CLI Version 7.0 Product Guide*

EMC documentation can be viewed and downloaded from the EMC Powerlink® website.