

Symmetrix Configuration Management



© 2009 EMC Corporation. All rights reserved.

Welcome to Symmetrix Configuration Management.

Revision History

Rev Number	Course Date	Revisions
1.0	June, 2009	Complete

Copyright © 2009 EMC Corporation. All rights reserved.

These materials may not be copied without EMC's written consent.

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, EMC ControlCenter, AdvantEdge, AlphaStor, ApplicationXtender, Avamar, Captiva, Catalog Solution, Celerra, Centra, CentraStar, ClaimPack, ClaimsEditor, ClaimsEditor, Professional, CLARAlert, CLARiiON, ClientPak, CodeLink, Connectrix, Co-StandbyServer, Dantz, Direct Matrix Architecture, DiskXtender, DiskXtender 2000, Document Sciences, Documentum, EmailXaminer, EmailXtender, EmailXtract, enVision, eRoom, Event Explorer, FLARE, FormWare, HighRoad, InputAccel, InputAccel Express, Invista, ISIS, Max Retriever, Navisphere, NetWorker, nLayers, OpenScale, PixTools, Powerlink, PowerPath, Rainfinity, RepliStor, ResourcePak, Retrospect, RSA, RSA Secured, RSA Security, SecurID, SecurWorld, Smarts, SnapShotServer, SnapView/IP, SRDF, Symmetrix, TimeFinder, VisualSAN, VSAM-Assist, WebXtender, where information lives, xPression, xPresso, Xtender, Xtender Solutions; and EMC OnCourse, EMC Proven, EMC Snap, EMC Storage Administrator, Acartus, Access Logix, ArchiveXtender, Authentic Problems, Automated Resource Manager, AutoStart, AutoSwap, AVALONidm, C-Clip, Celerra Replicator, CLARevent, Codebook Correlation Technology, Common Information Model, CopyCross, CopyPoint, DatabaseXtender, Digital Mailroom, Direct Matrix, EDM, E-Lab, eInput, Enginuity, FarPoint, FirstPass, Fortress, Global File Virtualization, Graphic Visualization, InfoMover, Infoscapes, MediaStor, MirrorView, Mozy, MozyEnterprise, MozyHome, MozyPro, NetWin, OnAlert, PowerSnap, QuickScan, RepliCare, SafeLine, SAN Advisor, SAN Copy, SAN Manager, SDMS, SnapImage, SnapSure, SnapView, StorageScope, SupportMate, SymmAPI, SymmEnabler, Symmetrix DMX, UltraFlex, UltraPoint, UltraScale, Viewlets, VisualSRM are trademarks of EMC Corporation.

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

Course Objectives

Upon completion of this course, you will be able to:

- Provide an overview of Symmetrix Configuration and Device Masking
- Create and Map Symmetrix Devices
- Describe how to set Symmetrix Metrics and Device Attributes, manage Dynamic RDF and Device Pools
- Map and Mask Devices using Autoprovisioning
- Migrate devices non-disruptively using Virtual LUNs
- Describe Virtual Provisioning Concepts
- Manage Virtual Provisioning with Solutions Enabler
- Monitor Virtual Pools
- Use the Symmetrix Management Console to perform Configuration Management

The objectives for this course are shown here. Please take a moment to read them.

Suggested Course Agenda (Your Speed May Vary)

- Day 1
 - Course Introduction
 - Configuration and Device Masking overview
 - Device Creation and Mapping
 - Symmetrix and Device Attributes
 - Autoprovisioning Groups
 - Explore Solutions Enabler and Symmetrix Environment (Lab)
 - Create and Map Devices (Lab)
 - Dynamic RDF Groups and Pairs (Lab)
- Day 2
 - Virtual LUN Technology
 - Virtual Provisioning (VP) Concepts
 - VP with Solutions Enabler
 - Autoprovisioning Groups (Lab)
 - Virtual LUN Migration (Lab)
- Day 3
 - Replication with Thin Devices
 - Monitoring Thin Pools
 - Create Thin and Data Devices and Thin Pools (Lab)
 - Map and Mask Thin Devices and Drain Data Devices (Lab)
 - Miscellaneous VP Operations (Lab)
 - Replication with Thin Devices (Lab)
- Day 4
 - Configuration with Symmetrix Management Console (SMC)
 - Monitoring Thin Pools (Lab)
 - Symmetrix Configuration with SMC (Lab)

Reference Material

- All references to documentation refer to EMC Solutions Enabler Version 7.0 documentation
- All references to the Array Control Guide refer to the Symmetrix Array Controls CLI Version 7.0 (P/N 300-002-940 Rev A06)

EMC²

where information lives

Module 1: Configuration and Device Masking Overview

Upon completion of this module, you will be able to:

- Describe the hardware components in the host to Symmetrix storage I/O path
- Identify the available tools for configuration and mapping
- Describe the CLI command structure for configuration
- List the methods for device masking on Symmetrix DMX and V-Max

The objectives for this module are shown here. Please take a moment to read them.

Terminology

- Symmetrix ID (*sid*)
- Physical Device (*Pdev*)
- Logical Device (*Ldev*)
- Symmetrix Device (*Dev*)

Symmetrix
Hardware



- Symmetrix Frame
- Physical Disk Drive
- Hyper-volumes (Splits)
- Symmetrix Logical Volume (*SLV*)

Solutions
Enabler



Depending on your point of view, a Symmetrix device can have a number of different names.

Host View of the Symmetrix

- The Symmetrix emulates disk drives
 - Open Systems Host simply see the Symmetrix as one or more FBA SCSI disk drives
 - Mainframe systems see the Symmetrix as a Logical Control Unit and one or more CKD disk drives
- The host has no knowledge of the Symmetrix internal configuration
- EMC provided integration tools provide visibility and control

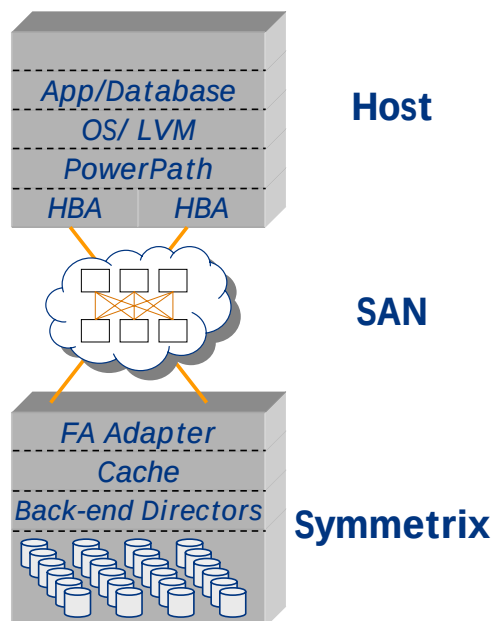


From a host's perspective, the Symmetrix is simply seen as one or more FBA or CKD devices.

Standard SCSI commands such as SCSI INQUIRY and SCSI READ CAPACITY return low-level physical device data, such as vendor, configuration, and basic configuration, but have very limited knowledge of the configuration details of the storage system. Knowledge of Symmetrix specific information such as director configuration, cache size, number of devices, mapping of physical-to-logical, port status, flags, etc. require a different set of tools, and that is what Solutions Enabler is all about.

IO Stack Open Systems

- Host
 - Application and/or Database
 - Operating System
 - Logical Volume Manager
 - Path Management Software (e.g. PowerPath)
 - HBA
- Storage Area Network (SAN)
 - Fibre Channel or TCP/IP
 - Switches & Cables
- Symmetrix
 - Front-end Director
 - Symmetrix cache
 - Back end director
 - Physical disk



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 4

Mercer Road
EMC Education Services

The diagram above illustrates the IO stack of a typical open systems host. Starting for the top, the host initiates a read or write operation. The I/O operation is passed by the I/O handlers in the operating system to the Logical Volume Management layer. Path management software such as PowerPath is optional but is part of most open systems environments. It performs two functions: load balancing and path failover.

While it is possible to connect a HBA directly to a Symmetrix front-end director port, more likely, you will be connected through a Storage Area Network or a SAN. A SAN consists of one or more interconnected switches. SANs provide greater connectivity by allowing more than one host to share the same front-end port on the Symmetrix.

In the Symmetrix, devices are presented to either a Fibre Channel or iSCSI front-end director and assigned a channel addresses. When more than one host is connected to the same front-end port auto provisioning is configured to restrict which host has access to which specific volumes.

The front-end directors were designed to support a large number of SCSI variants, therefore each port must be configured to support a subset of the SCSI and Fibre Channel protocol required for specific hosts.

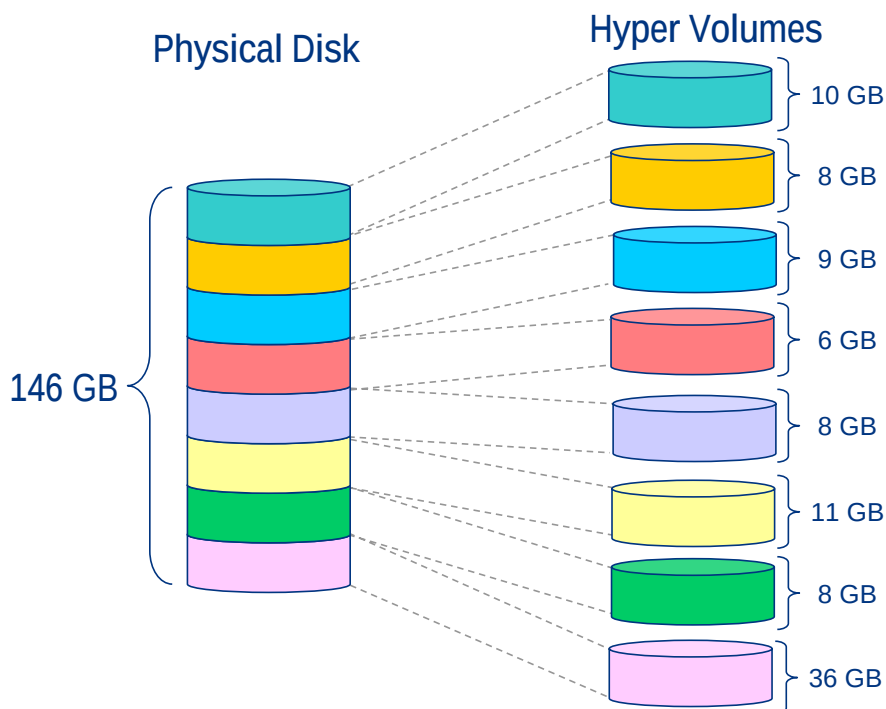
If the I/O request was a write, it passes through the front end adapter and through Symmetrix cache, and later directed through the back end disk adapters (DAs) to its final resting place on physical disk.

Symmetrix Logical Volumes

- Logical abstraction of a disk drive
 - Industry term is LUN – Logical Unit
 - EMC terms often used are hyper-volume, slice, split, device, or volume
 - Assigned a Volume Identifier
- Enginuity maps Logical Volumes to locations on Physical Disk on back-end
- Symmetrix Logical Volumes are made available to a host
 - Physical Connectivity
 - Front-end Channel Address
 - Device Masking

One of the most confusing aspects for new customers starting to use the Symmetrix is the number of different terms that are used to describe the same thing. Unfortunately, we've been in the storage business for so long that we've invented as many words for physical storage as eskimos are supposed to use for describing snow.

Physical Disk and Hyper Volumes



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 6

Mercer Road
EMC Education Services

The software used to “split” physical disks into volumes is called Hyper Volume Extension. Symmetrix physical disks are split into logical Hyper Volumes. Hyper Volumes (disk slices) are then defined as Symmetrix Logical Volumes (SLV). SLVs are internally labeled with hexadecimal identifiers (0000-FFFF). The maximum number of host addressable logical volumes per Symmetrix configuration is 64,000.

While “Hyper Volume” and “split” refer to the same thing (a portion of a Symmetrix physical disk), a “Symmetrix Logical Volume” is a slightly different concept. A Symmetrix Logical Volume is an abstraction of a disk drive that is presented to a host via a Symmetrix channel director port. As far as the host is concerned, the Symmetrix Logical Volume is a physical drive.

Symmetrix Logical Volumes are defined in the Symmetrix Configuration (BIN File). From the Symmetrix perspective, physical disk drives are partitioned into hyper volumes. A hyper volume could be used as an unprotected Symmetrix logical volume, a mirror of another hyper volume, a Business Continuance Volume (BCV), a member for RAID 5 or RAID 6 volume, a remote mirror using SRDF, and other uses.

Volume Table of Contents (VTOC) on disk are used to map logical volumes to physical disks. These data structures are created during initial installation.

- Maximum hyper volumes per physical disk varies with software version - currently 255 maximum
- Hyper volumes can be of variable size

Volume Size Metrics

- A Symmetrix Logical Volume is an emulation of a Physical Disk and uses similar terminology

- Sector

- (16) 512 byte Block = 8 KB

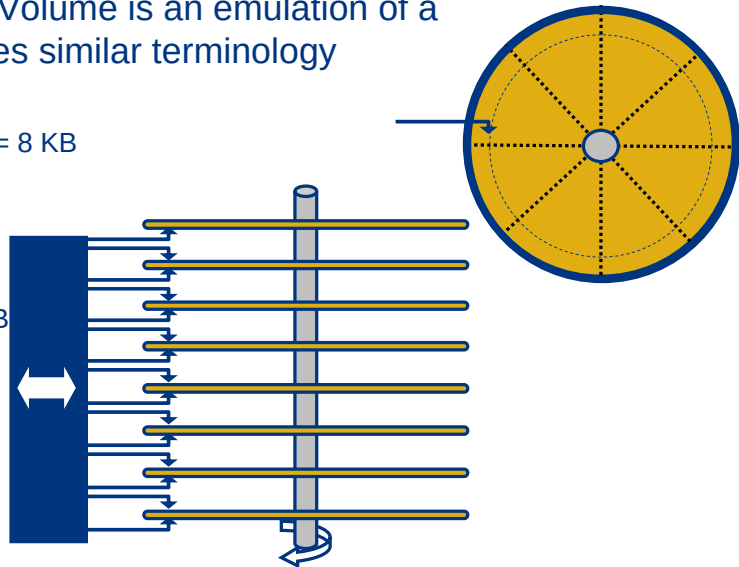
- Track (R/W Head)

- (8) Sectors = 64 KB

- Cylinders

- (15) Tracks = 960 KB

- Volume sizes are typically specified in Cylinders



9000 Cyl Device = 8.8GB $(9000 \times 15 \times 8 \times 16 \times 512 = 8,847,360,000)$

Largest Volume w. Enginuity 5874 = 262668 cylinders = 245760 MB = 240 GB

© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 7

 Mercer Road
EMC Education Services

Host I/O operation are managed by the Enginuity operating environment, which runs in the Symmetrix I/O subsystem (channel directors and disk directors). Because each of the physical disks are indirectly seen as part of the I/O protocol, Symmetrix devices are presented to the host with the following configuration or emulation attributes:

- Each device has **N** cylinders. The number is configurable (blocks ÷ 960)
- Each cylinder has **15** tracks (heads)
- Each device track in a fixed block architecture (FBA) has **128** blocks of **512** bytes (64K)
Note: prior to DMX3, the track size for FBA devices was 32K
- Mainframe hosts use Count Key Devices (CKD) uses variable block sizes.

Maximum Volume size that can be configured on a V-Max is 262688 cylinders. If host applications require larger volumes, multiple volumes can be combined to form a metavolume.

Data Protection Options for Symmetrix

Option	Characteristics	Protection	Performance	Cost
RAID 1	- Write to two separate physical drives - Read from single drive – DMSP	Higher	Fastest	Low Cost
RAID 5	- Parity based protection - Striped data and parity – 3+1 and 7+1 Configurations	High	Fast Read Good Write	Lower Cost
RAID 6	- Two parity drives – 6 + 2 and 14 + 2 - Data Availability is primary - Performance is a secondary consideration - New with Enginuity 5772	Highest	Fast Read Fair write	Lower Cost
Unprotected	Not recommended			

© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 8


 Mercer Road
 EMC Education Services

RAID 5 is based on the industry standard algorithm and can be configured with three data and one parity, or seven data and one parity. While the latter will provide more capacity per \$, there is a greater performance impact when in degraded mode where a drive has failed and all surviving drives must be read in order to rebuild the missing data.

RAID 6 is focused on availability. With the new larger capacity disk drives, rebuild times may take multiple days increasing the exposure to a second disk failure.

Random read performance is similar across all protection types, assuming you are comparing the same number of drives. The major difference is write performance. With mirrored devices for every host write there are two writes on the backend. With RAID 5, each host write results in two reads and two writes. For RAID 6, each host write results in three reads and three writes.

Other data protection schemes include remote replication using SRDF.

Comparison of RAID Protection Schemes

- Performance of reads similar across all protection types
 - If you are comparing the same number of drives!
- Major difference is with random write performance
 - Mirrored: 1 Host Write = 2 Writes
 - RAID 5: 1 Host Write = 2 Reads + 2 Writes
 - RAID 6: 1 Host Write = 3 Reads + 3 Writes
- Cost is an important factor
 - RAID 5/6 are best at 12.5% or 25% protection overhead
 - RAID 1 has a 50% protection overhead

Protection	Random Write Performance	Resiliency
Mirrored	Best	Better
RAID 5	Better	Good
RAID 6	Good	Best

Most RAID protection schemes offer similar read performance characteristics as long as the comparison takes a similar number of drives into account. However, write performance characteristics of different RAID types is quite different as the table above shows.

RAID-1 devices exhibit the best write performance. RAID-6 volumes offer the best protection against drive failure.

Optimal Volume Configuration

- Maximum number of hyper volumes on a physical drive with Enginuity 5874 is 512
 - Up from 255
- Generally, fewer larger hypers will give better overall system performance
 - There is system overhead to manage a logical volume, so it makes sense that more logical volumes could lead to more overhead
- Frequently legacy hyper size is carried forward because of migration strategy
 - However the same hyper size that made sense 10 years ago on a 9 GB disk drive, doesn't make sense on a 1 TB disk drive today

Prior to Enginuity 5874 the maximum numbers of hyper volumes permitted on a disk was a user settable parameter. With 5874, this number is fixed at 512.

This does not mean that it is appropriate to configure 512 hypers on a physical volume. Fewer larger volumes on a physical disk leads to better performance, since there is less contention for the resources on the physical disk.

Symmetrix Configuration Tools

- Initial configuration is created from the Symmetrix Service Processor to build an IMPL.bin file
- The “bin” file contains configuration information for a Symmetrix
 - Physical hardware configuration
 - Directors
 - Memory
 - Physical Drives
 - Logical storage configuration
 - Emulation (CKD or FBA)
 - Data protection (RAID type)
 - Director flags
- Subsequent changes are made on-line
 - EMC can update the IMPL.bin
 - Users can make changes on-line
 - Solutions Enabler CLI
 - Symmetrix Management Console (SMC)



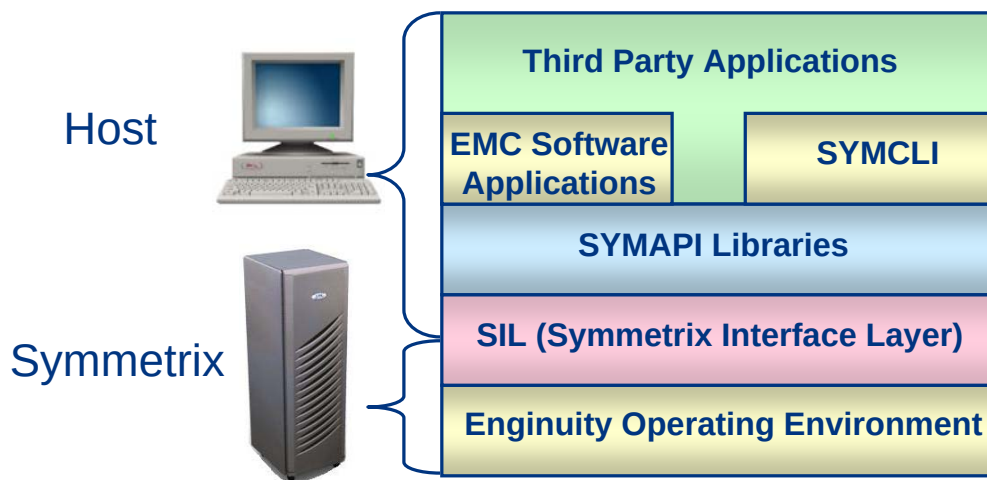
The Symmetrix is configured using a static configuration file called the IMPL.bin. The file is created initially using the SymmWin software from the Service Processor and loaded into each director in the Symmetrix. When modifying a configuration, the current IMPL.bin file is pulled from the Symmetrix and edited using Symmwin.

SymmWin is an EMC written graphical-based application for managing a Symmetrix. Capabilities include:

- Building and modifying system configuration files (IMPL.bin)
- Issuing Inlines commands, diagnostic, and utility scripts
- Monitoring performance statistics
- Automatically performs periodic error polling for errors and events. Certain errors will cause the service processor to “Call Home”.

SymmWin runs locally on a Symmetrix Service Processor or on a standalone PC. Running on the service processor allows communications with an operational Symmetrix. Running it on a standalone system allows you to build a new configuration or view and modify an archived configuration file.

Solutions Enabler Integration with Enginuity



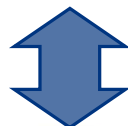
- SYMCLI commands are built on top of SYMAPI library functions
 - Use system calls that generate low-level I/O SCSI commands that are sent to the Symmetrix

This illustrates the software layers and where each component resides.

EMC's Solution Enabler APIs are the storage management programming interfaces that provide an access mechanism for managing the Symmetrix. They can be used to develop storage management applications. SYMCLI resides on a host system to monitor and perform control operations on Symmetrix arrays. SYMCLI commands are invoked from the host operating system command line (shell). The SYMCLI commands are built on top of SYMAPI library functions, which use system calls that generate low-level I/O SCSI commands to the storage arrays.

EMC Solutions Enabler Introduction

- Symmetrix Command Line Interface (SYMCLI)
- Provides a host with a comprehensive command set for managing a Symmetrix storage environment
 - Invoked from the host OS command line
 - Scripts that may provide further integration with OS and application
- Separate components licenses
- Security and access controls
 - Monitor only
 - Host-based and user-based controls

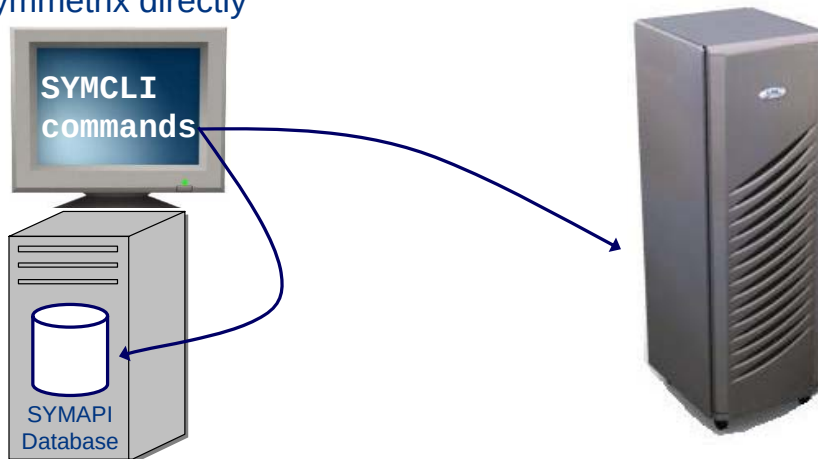


- ✓ Detailed Configuration Information
- ✓ Status
- ✓ On-line Configuration Changes
- ✓ Performance
- ✓ Control

The SYMCLI configuration change command, `symconfigure`, is used to perform control operations on Symmetrix® arrays and the array devices and ports. Some of the Symmetrix array controls include setting how many hypers per disk are allowed, and 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.

SYMAPI Database

- Symmetrix configuration and status info resides in SYMAPI database on the management host
 - Improve efficiency
 - Commands either act on information in the database or query the Symmetrix directly



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 14

Mercer Road
EMC Education Services

To reduce the number of inquiries from the host to the storage arrays, configuration and status information is maintained in a Symmetrix host database file called the Symmetrix configuration database (default file name: `symapi_db.bin`).

Configuration Manager Overview

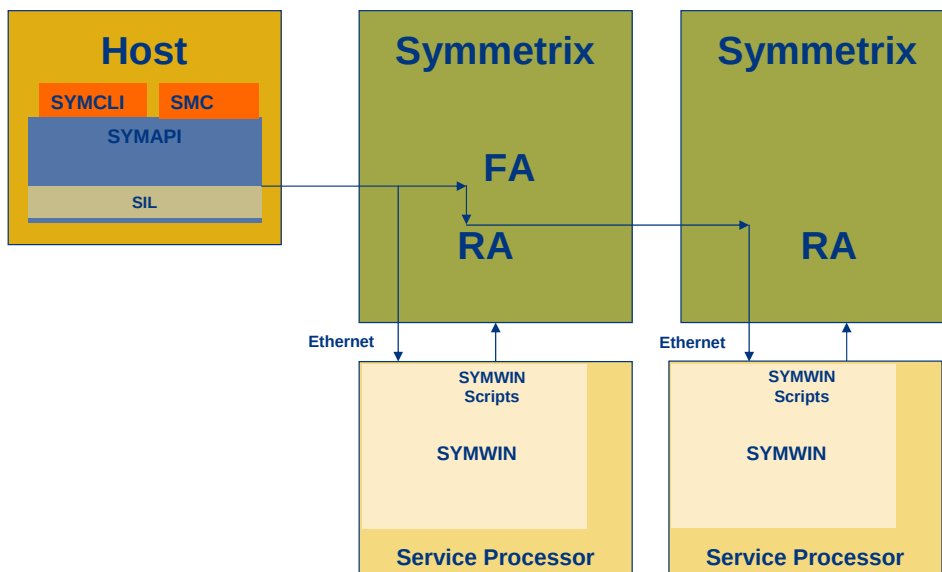
- Execute using the **symconfigure** SYMCLI command
- Performs configuration changes pertaining to:
 - Creation and attribute modification of all Symmetrix devices including save devices, spare devices and metadevices
 - SRDF group and device level characteristics
 - Mapping of devices to front end ports
 - Setting Symmetrix metrics
 - Setting front end port attributes

The SYMCLI Configuration Change Component, frequently referred to as the Config Manager is invoked using the **symconfigure** command. It can also be invoked through the Symmetrix Management Console GUI.

Config Manager is capable of configuration operations in the Symmetrix. A few SRDF related configuration activities cannot be performed by Config Manager. These include:

- Dynamic RDF group and pair creation and deletion which can be done with the **symrdf** command
- Modification of dynamic RDF group parameters such as **Prevent Automatic RDF Link Recovery** which can be set using the **symrdf** command
- Modify the **RAS online upon Power On** parameter which has to be set through the Symmetrix Service Processor using **Symwin**.

Architecture



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 16

Mercer Road
EMC Education Services

The Config Manager architecture allows it to run Symwin scripts on the Symmetrix service processor. Configuration change requests are generated either by the `symconfigure` SYMCLI command or a SYMAPI library call generated by a user making a request through the Symmetrix Management Console (SMC) GUI.

These requests are converted by SYMAPI on the host to Symmetrix syscalls and transmitted to the Symmetrix through the channel interconnect. The Symmetrix front end routes the requests to the service processor, which invokes Symmwin procedures to perform the requested changes to the Symmetrix.

Since these scripts are the same ones that a Customer Services Engineer would employ to configure the Symmetrix, Config Manager is able to do almost everything that is possible through Symwin scripts.

Operation Classes

- Array Wide Operations
- Device Operations
- Metadevice Operations
- Device Mapping
- Device Pool Management
- RDF Configuration (Older Symmetrix Arrays)
- RDF Group parameters
- Front-end port attributes

Configuration Manager operations can be divided into several classes. A detailed listing of the actions available under each class is described in the Symmetrix Array Controls Guide.

Array wide operations allow the setting of Symmetrix metrics.

Device operations comprise creation and deletion of devices, modification of device attributes and binding and unbinding of Thin devices to pools.

Metadevice operations allow the formation and dissolution of metavolumes.

Mapping makes a device available to a Symmetrix front-end port

Device pools operations allow for the management of Thin pools, Snap pools and SRDF DSE pools

RDF Configuration actions are more meaningful with the older Symmetrix arrays where static RDF groups are more prevalent

RA group parameters relate to RDF/A groups and DSE pools for SRDF/A

Front-end port attributes are set to cater to the needs of different vendors' operating systems

Preparation for Making Configuration Changes

- Verify configuration changes can be made safely
 - **symconfigure verify**
- Check for free disk space
 - **symconfigure list -freespace**
- Ensure all critical data is preserved by using protected devices
- Consider impact on I/O
 - To make devices not ready use: **symdev not_ready <SymDev>**

Before making configuration changes, it is important to know the current Symmetrix configuration.

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

```
symconfigure verify -sid SymmID
```

will return successfully if the Symmetrix is ready for configuration changes

Free physical disk space can be checked using the command:

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

Use protected devices for storing data. Check the product documentation to understand the impact that a configuration change operation can have on host I/O.

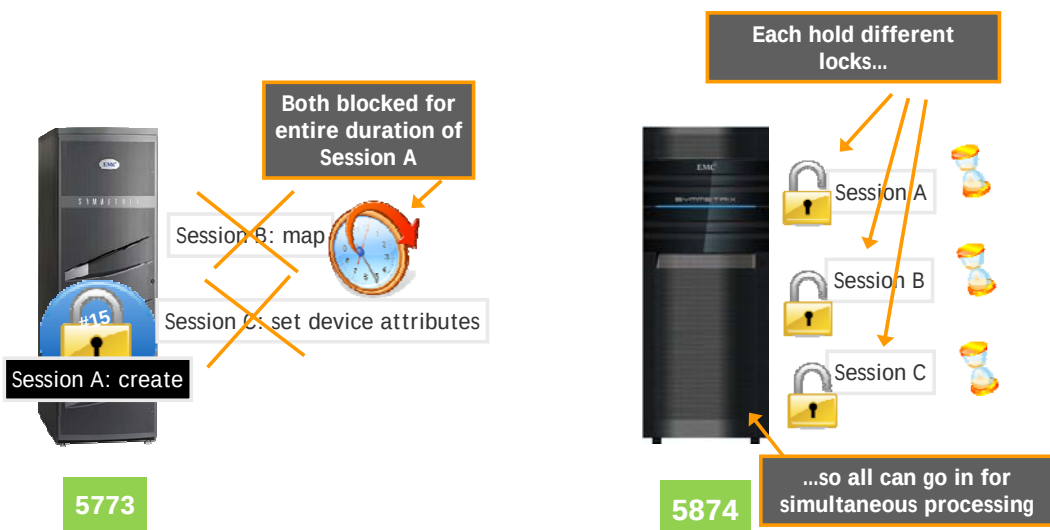
Submission of Configuration Change Requests

- Create command file
 - Multiple commands separated with semi-colon
- Three ways of comand submission
 - Submit file containing commands as a parameter to **symconfigure**
 - e.g. **symconfigure -sid 123 -file myfile commit**
 - Enclose commands within quotes following the **-cmd** option
 - e.g. **symconfigure -sid 123 -cmd "delete dev 0015;" commit**
 - On Unix systems enter the commands via *stdin* on the screen
 - **symconfigure -sid 123 -noprompt prepare <<EOF
dissolve meta dev 002;
EOF**

Configuration change requests are placed in a command file. The syntax for these commands is described in Chapter 1 of the Symmetrix Array Control CLI Product Guide.

Prior to Enginuity 5669 only one class of commands could be submitted for execution at one time. Though that restriction does not exist today any more, changes to dynamic RDF, Save pools and protected expansion of striped metavolumes can still not be mixed with other class operations in the same command file.

Concurrent Command Execution



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 20

Mercer Road
EMC Education Services

Before 5874 a configuration change request acquired Symmetrix lock 15 and prevented other changes from being made until the first was completed. In 5874 configuration changes need not be done one at a time.

In 5874 a change operation only takes the resources it needs to do its job. The old locks 15 and 9 have been replaced with four separate locks described later. The new locks reserve system resources with greater granularity and for shorter duration on the Symmetrix V-Max. With Solutions Enabler 7.0 on a DMX running 5773 or earlier, locks will still be taken one at a time and no parallelism is allowed.

Configuration Locks in 5874

- Director lock
 - One for every director
 - Taken when changes affecting the director are performed
- Front-end device lock
 - One per device
 - Taken when changes affecting a device are performed
- Backend device lock
 - One per device
 - Taken when changes are undertaken in the backend
- Configuration lock locks the IMPL file
 - Taken when the revised IMPL file has to be reloaded
 - One per Symmetrix V-Max

This information shown here is not exposed to the user and hence not documented in the product guide. There are four kinds of locks that can be taken during a configuration change session. The granularity of the locks allow operations that do not take the same locks to run in parallel.

Director locks are taken by changes when a director is affected. A mapping operation will lock the director to which a device is being mapped. However, a different device being mapped to a different director will not take the same locks and could therefore be mapped in parallel.

The front end device lock is taken when the front end of the device, such as meta-formation or device mapping is being undertaken.

The backend device lock is taken when the back end is being affected. An Optimizer swap affects the back end only.

Changes that require a reloading of the IMPL.bin file take the static configuration lock.. This lock prevents other configuration changes that require IMPL changes. Other configuration changes can happen concurrently if they need device locks or director locks.

There is a **CE Config Lock**, that is set from the Service Processor, that acts like the old lock 15. It prevents any other configuration changes from happening on the Symmetrix.

Parallel Command Execution

Host A:
map devices to port 7E:0



Host B:
create new devices



Host C: map devices to port 8F:0



Host D: map devices to port 8F:0



	Device Front End Lock	FBA Director Lock	Device Back End Lock	Static Lock
FBA Map / Unmap	YES	YES	NO	NO
Create Device	YES	NO	YES	YES

© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 22

Mercer Road
EMC Education Services

In the example shown a group of devices being mapped to a director takes the front end lock for the devices involved and a lock on the director to which the devices are being mapped.

A device creation operation takes the static configuration lock, the device front and back end locks for the devices being created. Since the devices being created are not the ones being mapped, the two changes can occur in parallel.

If however, the device creation step also included a mapping step that mapped the newly created devices to the same director to which the first set of devices are being mapped, the two actions could not happen in parallel.

On the right hand side are two hosts competing for the same front end director port. As a result one of the hosts that issues the second request for the front end director resources will fail to complete the change.

Querying Configuration Change Sessions

```
# 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 : api1051
```

```
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.
```

```
}
```

Configuration change sessions can be viewed using the **symconfigure query** command. If there are multiple sessions running all session details are shown.

Termination of Configuration Change Commands

- Configuration change sessions can be terminated prematurely using the **abort** command
- Premature termination is only possible before the point of no return
- Syntax
 - **symconfigure -sid <SymmID> abort -session_id <SessID>**

In rare instances, it might become necessary to abort configuration changes. This can be done with the symconfigure abort command as long as the point of no return has not been reached. Aborting a change that involves RDF devices in a remote array might necessitate the termination of changes in a remote array.

```
# symconfigure -sid 207 -session_id 39682 abort -noprompt
A Configuration Change abort is in progress. Please wait...
A Configuration Change operation is in progress. Please wait...
```

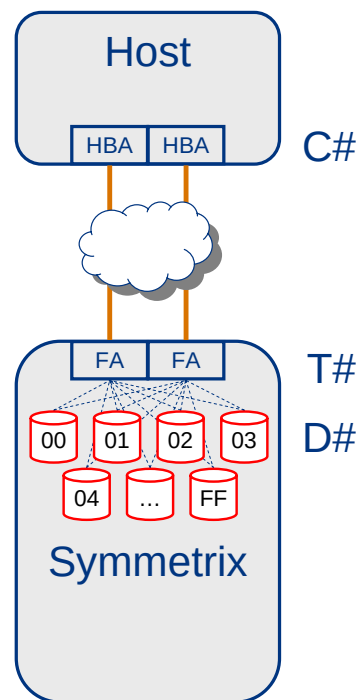
```
Looking for an existing configuration session.....Established.
The session changes are in the class of: Creating new symdevices
{
  create dev count=1, size=2200 cyl, emulation=FBA, config=2-Way Mir,
  mvs_ssid=a;
}
```

```
The Application that initiated the configuration change : SYMCONFIGURE
The Host that initiated the configuration change       : api1051
The Process ID that initiated the configuration change : 12382
The session length                                   : 18 secs
Aborting configuration changes.....Aborted.
Terminating the configuration change session.....Done.
```

The configuration change session has been aborted.

Mapping a Device to a Port using Config Manager

- Symmetrix Devices are made available on an FA port by assigning them a channel address
 - Referred to as mapping
- Channel address is used by the host to access device
 - Often reflected in the c#t#d# device naming convention
 - Example channel address 0003 may be seen as c1t0d3
- Devices are mapped to 2 or more ports and managed by host based path management software for
 - Higher Availability
 - Load balancing



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 25

Mercer Road
EMC Education Services

A Symmetrix can have over 64000 devices configured. Not all devices are accessed by every front-end port. Instead, specific devices are “mapped” to specific ports by assigning a channel address. Host systems discover and access Symmetrix devices using these Channel Addresses. For open systems hosts, the Channel address is the SCSI ID. Normally a host uses a combination of the Controller, Target, and Logical Unit Number to address a disk device. The Controller number is the Host Bus Adapter, the Target is the port on the Storage System and the Logical Unit Number is the Channel Address we assign.

The reverse of mapping a device is **unmapping** a device. Unmapping can become necessary prior to a device being converted from one type to another e.g. a standard to a meta member. Before the device is unmapped it has to be set not ready. The unmap action will fail if the device is R/W enabled.

Connecting a Host to an older Symmetrix (DMX)

- HBA and device drivers must be installed and configured
- SAN connection between the Host Bus Adapter (HBA) and the Symmetrix front-end director port
 - Physical cable connection
 - Logical connection (zoning)
- Director Port Characteristics
 - SCSI and Fibre Channel operating parameters
 - Host Operating Systems Specific
- Device Mapping
 - Make a device visible via the front end adapter port by assigning it a target ID and logical Unit Number
- Device Masking
 - A single FA port can be shared by many hosts
 - Controls access so specific hosts see specific devices
 - Masking information maintained in Volume Logix Database

Before connecting an open systems host to the Symmetrix the following questions should be answered:

- Which host is going to connect to which port
- What are the operating systems and versions of the hosts
- Number, type, and firmware levels for Host Bus Adapters (HBA)
- Is PowerPath or other multi-pathing failover software used
- How many, what protection, what size volumes are required
- Performance considerations (e.g. faster disks should be picked for high performance applications).

On the host side, the host bus adapter (HBA) has to be configured with the correct drivers. Multi-pathing software if present needs to be set up on the host.

The physical SAN connection between the host and the Symmetrix consists of cables and SAN equipment such as Fibre channel switches. Logical zones are needed to establish a connection between the host bus adapter and the Symmetrix front end ports.

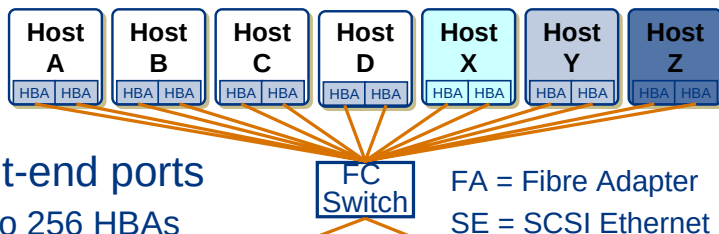
On the Symmetrix side the front end adapter (e.g. FA) needs to be cabled to the SAN and zoned such that the host HBA and the front end adapter (e.g. FA) are in the same zone. Zoning can be done using software from the SAN vendors.

The characteristics of the front end adapter port, to which the HBA connects, need to be appropriately set so the host operating system can access the Symmetrix devices. Device mapping permits a device to be accessible through a front end port. Config Manager is the appropriate tool to perform both of these tasks..

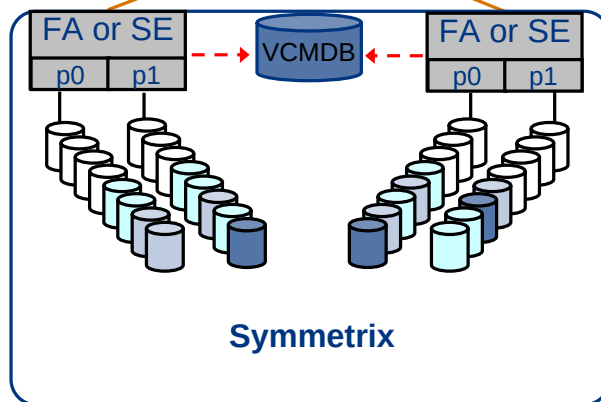
Device masking permits only a subset of devices that are mapped to a port to be visible to an HBA. This feature allows multiple hosts to share the same Symmetrix front end port without encroaching on another host's devices. Device masking is performed using the masking commands in Solutions Enabler `symmask` and `symmaskdb`

Masking on a Symmetrix DMX

- Device masking allows multiple hosts to effectively share the same front-end ports
 - FA port can “see” up to 256 HBAs



- Restrict access to specific host and/or host clusters
- Implemented in the Symmetrix with Volume Logix
 - Fibre Channel
 - iSCSI



© 2009 EMC Corporation. All rights reserved.

Configuration and Device Masking Overview - 27

Mercer Road
EMC Education Services

Storage Area Networks provide a fan-out capability where it is likely that more than one host is connected to the same Symmetrix port. The actual number of HBAs that can be configured to a single port is operating system and configuration dependent but fan-out ratios as high as 256:1 are currently supported. Reference the support matrix for specific configuration limitations.

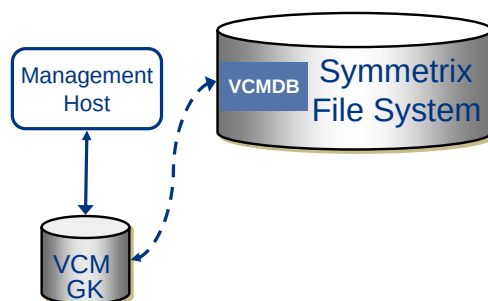
When several hosts connect to a single Symmetrix port, an access control conflict can occur because all hosts have the potential to discover and use the same storage devices. However, by creating entries in the Symmetrix's device masking database (VCMDB), it is possible to control the volumes “seen” by a host.

Device Masking is independent from zoning but zoning and masking are typically used together in an environment. Zoning provides access control at the port level. It establishes a logical connection between the host bus adapter and port on the storage system. Device masking allows a subset of volumes mapped to a port to be visible to the host bus adapter.

With Fibre Channel, Device Masking uses the UWWN (Unique Worldwide Name) of Host Bus Adapters and a VCM database device. In iSCSI, the iSCSI Qualified Name (IQN) is used. Regardless of the protocol, the concepts are the same. The device-masking database (VCMDB) on each Symmetrix unit specifies the devices that a particular WWN or IQN can access through a specific Fibre port.

Volume Logix Database - VCMDB

- Database types are Symmetrix model specific and sized to handle the maximum number of logical volumes
 - Originally a separate Symmetrix Logical Volume
 - Starting with the DMX-3, it is stored in the Symmetrix File System
- VCM DB is maintained using Solutions Enabler symmask and symmaskdb commands
- Solutions Enabler accesses the VCM database using a VCM gatekeeper device
 - VCM gatekeeper is host accessible
 - The SFS is not directly accessible
 - VCM GK only needs to be mapped to the management host

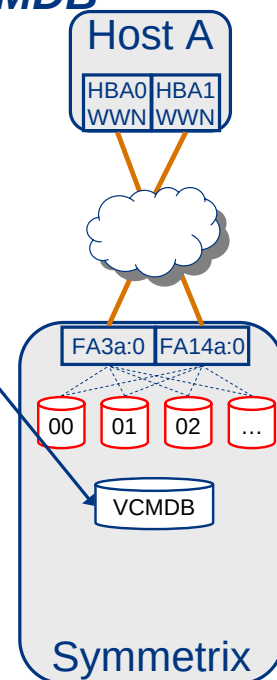


The Volume Logix Database persistently maintains the device masking information. Originally the database was located directly on a Symmetrix Logical Volume. On DMX-3 it is maintained in the Symmetrix File System (SFS). Rather than create the actual VCMDB device, today we create a VCM Gatekeeper device which is used by the Solutions Enabler to access the database on the SFS, as the SFS volumes are not host addressable. The VCM Gatekeeper is a 6-cyl device.

By default, the device masking VCMDB is accessible to all HBAs that log into 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 commands are installed. Beginning with Engenuity Version 5670, the VCMDB can be unmapped from any director that is not being used for masking control.

Connection Records Maintained in VCMDB

- Example using 2 dual port adapters
 - HBA0 WWN -> FA03a:0 - dev 000-010
 - HBA0 WWN -> FA14a:0 - dev 000-010
 - HBA1 WWN -> FA03a:0 - dev 000-010
 - HBA1 WWN -> FA14a:0 - dev 000-010
- Entries in the VCMDB define relationship between masked connections and devices
 - FA consults VCMDB to resolve access rights
- Same approach for both FC and iSCSI



Device Masking controls 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. These entries are sometimes called initiator records.

Each entry includes a host's HBA identity (WWN or iSCSI Qualified Name), 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 through that FA port.

Volume Logix is the brand name for the software in the Symmetrix that performs the device masking function. The capability is built into Enginuity but its use is optional.

Autoprovisioning Groups on Symmetrix V-Max

Starting with Symmetrix V-Max and Enginuity 5874

Autoprovisioning replaces Volume Logix as the way to mask Symmetrix storage

Initiator Group	- Fibre Channel Initiator / iSCSI name - Port Flags set on Initiator Group - FCID Lockdown per initiator
Port Group	- Front End Ports - A port can belong to multiple port groups - Ports must have the ACLX flag enabled
Storage Group	- Symmetrix logical volumes - A device can belong to more than one Storage group

An Initiator Group contains the world wide name or iSCSI name of a host initiator, also referred to as an HBA or host bus adapter. An initiator group may contain a combination of up to thirty-two, Fibre Channel initiators or eight, iSCSI names or a combination of both. There is a limit of eight thousand one hundred ninety-two, (8,192) initiator groups in a Symmetrix V-Max array. Port flags are set on an initiator group basis, with one set of port flags applying to all initiators in the group. However the FCID lockdown is set on a per initiator basis. An individual initiator can only belong to one Initiator Group.

However once the initiator is in a group, the group can be a member in another initiator group. It can be grouped within a group. This feature is called cascaded initiator groups, and is only allowed to a cascaded level of one.

A Port Group may contain any number of valid front end ports, FAs. Front end ports may belong to more than one port group. There is a limit of five hundred twelve (512) port groups. Before a port can be added to a port group the ACLX flag must enabled on the port.

A Storage Group may contain up to four thousand ninety-six, (4,096) Symmetrix logical volumes. A logical volume may belong to more than one storage group. There is a limit of eight thousand one hundred ninety-two, (8,192) storage groups.

Lecture Assignment 1-1

- Review the section on Configuration Control Operations in the Array Control Guide Version 7.0 (Page 30 -34)
- Answer the questions:
 - To which operation class do the following actions belong:
 - Addition and removal of spare devices
 - Binding and unbinding of thin devices
 - Setting thin pool attributes
 - What is the minimum microcode level needed to support
 - RAID-5 devices
 - RAID-6 devices
 - Adding and removing RDF mirrors

Module Summary

Key points covered in this module:

- Hardware components in the host to Symmetrix storage I/O path
- Tools for configuration and mapping
- CLI command structure for configuration
- Methods for device masking on Symmetrix DMX and V-Max

These are the key points covered in this module. Please take a moment to review them.

Module 2: Device Creation and Mapping

Upon completion of this module, you will be able to:

- Create and Delete Devices
- Form and Dissolve Meta volumes
- Map and Unmap Devices
- Set Port Characteristics
- Set Device Reservations



The objectives for this module are shown here. Please take a moment to read them.

RAID Virtual Architecture

- A Symmetrix volume may have up to 4 mirrors
- Prior to 5874 each local and remote mirror took up a mirror position
 - A locally mirrored volume with RDF protection occupied 3 mirror positions
 - A RAID-5 volume occupied 2 mirror positions
- RVA hides back end management of RAID groups from rest of Symmetrix
 - Regardless of RAID type all volumes occupy one local mirror position
 - RDF remote mirror(s) still occupy mirror position(s)
 - Concurrent and cascaded RDF volumes occupy 3 mirror positions

Each Symmetrix logical volume is allowed up to 4 mirrors. Prior to Enginuity 5874, local mirrors and parity RAID mirrors would occupy a second mirror position. This limited the number of remote mirrors and TimeFinder/Mirror BCVs these volumes could be joined to.

In 5874, TimeFinder/Mirror is no longer supported. Remote mirrors still occupy a mirror position but local mirrors no longer occupy a mirror position. Thus the 4 mirror limit is no longer a limiting factor.

Common Device Types

- 2-Way-Mir, RAID-5, RAID-6
- BCV (2-Way-Mir, RAID-5, RAID-6)
- RDF1 (2-Way-Mir, RAID-5, RAID-6)
- RDF2 (2-Way-Mir, RAID-5, RAID-6)
- RDF1-BCV (2-Way-Mir, RAID-5, RAID-6)
- RDF1-BCV (2-Way-Mir, RAID-5, RAID-6)
- VDEV, TDEV, DLDEV
- DATA, SAVE

Apart from unprotected devices, which are not recommended, Symmetrix volumes can be configured with RAID-1, RAID-5 or RAID-6 protection.

BCVs are device types that are used for local replication.

RDF volumes are used for remote replication

Virtual devices are used in TimeFinder/Snap. They are cache only devices and do not consume disk space.

Thin Devices are used for Virtual provisioning. They are cache only devices and do not consume disk space.

Diskless devices are used for cascaded R21s. They are cache only devices and do not consume disk space.

Save and Data devices hold the actual data for Virtual and Thin devices respectively.

Each of these devices can be created with the Configuration manager. Less commonly used devices, such as DRV devices have not been listed above.

Device Creation Examples

- Creation of RAID-6 device with 6+2 RAID protection

- **create dev count=4, size=1100, config=RAID-6, emulation=FBA, data_member_count=6;**



- Creation of dynamic RDF capable 2-Way Mirror in disk group 1

- **create dev count=8, size=1100, emulation=FBA, config=2-Way-Mir, disk_group=1, dynamic_capability=dyn_rdf;**



- Creation of a Virtual Device

- **create dev count=2, size=1100, emulation=FBA, config=VDEV;**

- Creation of a RAID-5 protected Save Device

- **create dev count=4, size=1100, emulation=FBA, config=RAID-5, data_member_count=7, attribute=savedev;"**

```
create dev count=n, size=Cylinders,
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], ]
[, disk_group=nnn, remote_disk_group=nnn]
[, binding to pool=PoolName
[preallocate size=n [MB |GB | CYL]
[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]]...];
```

Deletion of Devices

- Device must be free of BCV or Snap sessions
- Device must not be
 - Mapped to a front-end port
 - Part of an RDF consistency group
 - A source or target of a clone session
 - Held
 - A virtual device that is in use
 - An RDF device
 - A metamember
 - An SFS device
 - A SAVE device¹
 - The VCM database device
 - Masked by VCM
 - Bound if a thin device
- Syntax: **delete dev 0010:0020;**

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. There are restrictions on device deletions that are aimed at protecting the data on the devices or any devices that may have associations with that device. This is the reason behind not allowing the deletion of devices with Snap or BCV sessions.

The complete syntax for device deletion is:

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

Device Deletion Considerations

- Device Deletions can create gaps in numbering

```
# symdev list -all
```

```
Symmetrix ID: 000190104642
```

Device Name		Directors		Device		
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)
01A3	Not Visible	????:? 16B:C2	DRV	N/Grp'd	RW	898
01A4	Not Visible	????:? 01D:D2	DRV	N/Grp'd	RW	898
~~01A5~~	~~Not Visible~~	~~????:? 16C:D2~~	~~2-Way Mir~~	~~N/Grp'd~~	~~RW~~	~~2156~~
~~01A6~~	~~Not Visible~~	~~????:? 16D:C2~~	~~2-Way Mir~~	~~N/Grp'd~~	~~RW~~	~~2156~~
01A7	Not Visible	????:? 16D:D3	2-Way Mir	N/Grp'd	RW	2156

```
# symconfigure -sid 42 -nop -cmd "delete dev 1A5:1A6;" commit
```

```
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.
The configuration change session has successfully completed.
```

When devices are deleted, the device numbers they used to occupy disappear from the list of Symmetrix devices. Thus deletion of devices have the potential for creating noncontiguous device numbers in the Symmetrix.

Device Deletion Considerations - 2

- Gaps created by device deletions are not always filled by new devices
- After Deletion hole appears between 0A4 and 0A7

01A3	Not Visible	????? 16B:C2	DRV	N/Grp'd	RW	898
01A4	Not Visible	????? 01D:D2	DRV	N/Grp'd	RW	898
01A7	Not Visible	????? 16D:D3	2-Way Mir	N/Grp'd	RW	2156
01A8	Not Visible	????? 01C:D3	2-Way Mir	N/Grp'd	RW	2156

```
# symconfigure -sid 42 -nop commit -cmd "create dev count=4, size=1150,
emulation=FBA, config=2-way-Mir
```

- After Creation new devices end up at the bottom

01AB	Not Visible	***:* 16D:D1	2-Way Mir	N/Grp'd (M)	RW	1796
01AC	Not Visible	***:* 01D:C1	2-Way Mir	N/Grp'd (m)	RW	-
01AD	Not Visible	????? 01B:D2	2-Way Mir	N/Grp'd	RW	1078
01AE	Not Visible	????? 01A:C2	2-Way Mir	N/Grp'd	RW	1078
01AF	Not Visible	????? 01B:C1	2-Way Mir	N/Grp'd	RW	1078
01B0	Not Visible	????? 16C:C1	2-Way Mir	N/Grp'd	RW	1078

Although the configuration tool allows the deletion of existing devices, it does not allow the assignment of specific device numbers when new devices are being created. Symmwin uses internal algorithms for the best distribution and placement of devices in the Symmetrix and the user has no control over the placement or numbering of new devices. In the example shown, a gap was created in the Symmetrix device numbers after 1A4 due to the deletion of devices A5 and A6. However, a subsequent creation of 4 devices does not fill up the gaps left by the deletions.

Meta Device

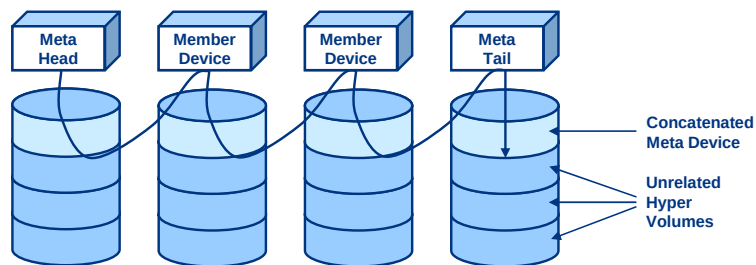
- Several symmetrix volumes presented to the front end as one device
- Two types
 - Concatenated
 - Striped

A meta device is a Symmetrix mechanism for defining a device larger than the current maximum hyper-volume size. You can concatenate existing devices to form a larger meta device that is presented to the host as a single addressable device.

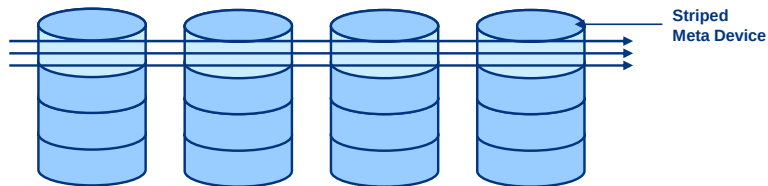
There are two kinds of meta devices - concatenated and striped

- On a concatenated meta device addressing of data continues to the end of a device before any data on the next device is referenced.
- On a striped meta device data on meta members is addressed in user-defined stripes or chunks instead of filling an entire volume first before addressing the next volume.

Meta Device Configurations



Concatenated Meta Device on Four Physical Disk Spindles



Striped Meta Device on Four Physical Disk Spindles

The meta head is the Symmetrix device that is recognized by the host and used for performing I/O.

Striped vs. Concatenated Meta

- Usually, striped meta volumes will perform better than concatenated ones
 - Because they reside on more spindles.
 - If there are not enough drives for all the meta members to be on separate drives consider concatenated.
 - If host striping is planned, concatenated meta volumes may be better
- Concatenated Meta Volumes can be placed on the same RAID group
 - Can create a reasonable emulation of a native large volume on Symmetrix DMX systems or Symmetrix V-Max systems
- Don't place striped meta volumes on the same RAID group (wrapped)
- For Virtual Provisioning environments, concatenated meta volumes work better
 - (Thin Meta Volumes)

Striped meta volumes perform better than concatenated meta volumes when there are enough spindles to support them. However, if the striping leads to the same physical spindle hosting two or more members of the meta volume, striping loses its effectiveness. In such a case, using concatenated meta volumes is better.

It is not a good idea to stripe on top of a stripe. Thus, if host striping is planned and meta volumes are being used, concatenated meta volumes are better.

In Virtual Provisioning environments it is advisable to use concatenated metavolumes

Metadevice Configuration Considerations

- Devices must be unmapped
- All members of a striped meta must have the same size and mirror protection
- Metadevices can be formed from
 - Virtual Devices
 - Thin Devices
- Members can be removed from the tail of concatenated metavolumes
- Members cannot be removed from striped metavolumes

Before a meta can be formed, the devices must be unmapped to guard against data loss.

Metavolumes can be formed from virtual and thin devices

Members can be removed from the tail of concatenated metavolumes but not from striped metavolumes

Metadevice Creation

- The traditional method, which still works, is to use the “form meta” syntax

form meta from dev 107, config=concatenated; add dev 108 to meta 107;

- Automatic meta device creation with SE 6.5.1 and 5773 if Symmetrix is set up to do so
- By default auto meta feature is disabled

```
# symcfg list -v -sid 42
```

```
.....
Auto Meta                               : Disabled
Minimum Auto Meta Size                  : 65521
Auto Meta Member Size                   : 0
Auto Meta Configuration                  : N/A
```

Metadevices can be created in either be formed using symconfigure or they can be automatically created using Solutions Enabler 6.5.1 or higher. The syntax for forming metavolumes is:

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

The stripe size parameter is not used for Enginuity versions 5669 and later. It is always 2 cylinders or 1920 KB.

Auto-meta Symmetrix settings

- **auto_meta**
 - can be enabled or disabled (ENABLE or DISABLE)
- **auto_meta_config**
 - striped or concatenated
- **auto_meta_member_size**
 - specifies default member size (MB or GB or cylinders) when feature is enabled
- **min_auto_meta_size**
 - threshold value that triggers a meta device creation

The explanation on auto-meta features is clearer in the man page for symconfigure than in the documentation.

The first two settings are self explanatory. The auto_meta_member_size is the default member size when metavolumes are automatically created.

The min_auto_meta_size specifies the size threshold that triggers auto meta creation. When users try to create a device greater than min_auto_meta_size, and auto_meta is enabled, a meta will be created. Possible values are between 0 and the maximum value from the table below (the default value is the same as the maximum value):

Enginuity	Max/Default cyl

5773	65521
5874+	262669

Enabling Symmetrix for Auto-Meta

```
set symmetrix auto_meta=ENABLE, min_auto_meta_size=65520,  
auto_meta_member_size=16380, auto_meta_config=concatenated;
```

- Setting specifies that
 - Any attempt to create a device greater than 65520 cylinders (~63 GB) will trigger an automatic meta device
 - Each metadvice member will be 16380 cylinders (~15.7 GB)
 - The meta devices will be concatenated
 - The stripe size for Enginuity 5669 and later (not indicated by the values shown) is always 2 cylinders (1920 KB)

This example shows how to set up a Symmetrix for autometa creation.

Dissolving a Metavolume

dissolve meta from dev 107;

- Removes the meta head
- Frees up the meta members
- Data on the meta volume is lost

Dissolving metavolumes frees up the members and makes the data unavailable to hosts that were accessing the data

Meta Volume Sizing Considerations

- Enginuity supports up to a 255 member meta
 - This represents a ~15 TB volume (Symmetrix DMX-3/4 systems)
 - Or a ~60 TB volume (Symmetrix V-Max systems)
- Reasonable sizes for meta member counts are:
 - 4, 8, 16 or 32
- Even numbers are preferred
 - Powers of 2 fit nicely into the Symmetrix system back-end configurations
- Getting enough I/O into a very large meta can be a problem
 - 32-way RAID 5 7+1 Meta Volume would need at least 256 I/Os queued to have 1 I/O per physical disk

Symmetrix Enginuity supports metavolumes with up to 255 members. As the largest volume size has gone up with each enginuity version, the size of the largest supported meta volume has gone up.

From a performance perspective it is best to use meta volumes with member counts that are powers of 2.

Steps to Map a Device to a Front End Port

- Devices can be mapped with the use of auto-provisioning groups on Symmetrix V-Max running 5874

Mapping with Config Manager works on all Symmetrix platforms

- Identify FA port to which the host receiving the new device is connected
 - **symcfg list -connections**
- Identify available addresses that the device could be assigned to
 - **symcfg list -available -address -sid <SID> -sa <XX> -p<YY>**
- Create a command file containing the mapping request
 - **map dev <xx> to dir <XX>:<YY>, lun=<LUN>**
- Execute the command file
 - **symconfigure -file <command file> -sid <SID> commit**

© 2009 EMC Corporation. All rights reserved.

Device Creation and Mapping - 17

Mercer Road
EMC Education Services

Before the newly created device can be used it has to be mapped to a front end port to which the receiving host is connected. For example, the output below indicates that the host DMX800SUN1 is connected to FA 2C port 0

```
# symcfg list -connections
Symmetrix ID : 000190300477
Symmetrix
```

Host

Director Port	Node Name	IP Address	HW Type	OS Name	OS Revision
FA-2C 0	DMX800SUN1	10.127.38.35	sun4u	SunOS	5.9
1	DMX800WIN1	10.127.38.33	INTEL	WinNT	5.2.3790

The next command shows that LUN numbers 41 and higher are available for assignment

```
# symcfg list -addr -avail -sid 77 -fa 2C -p 0
Symmetrix ID: 000190300477
```

Director			Device Name		Attr	Address		
Ident	Symbolic	Port	Sym	Physical		VBUS	TID	LUN
FA-2C	02C	0	0020	/dev/rdisk/c2t0d0s2	VCM	0	00	000
			00A9	/dev/rdisk/c2t0d1s2		0	00	001
			00EA	/dev/rdisk/c2t0d62s2		0	00	03E
			00EC	/dev/rdisk/c2t0d64s2		0	00	040
			-	AVAILABLE		0	00	041*

Unmapping Devices

- Devices must be write disabled (WD) or Not Ready before unmapping
- Unmapping can be done through **symaccess** auto-provisioning command on Symmetrix V-Max arrays

Mapping with Config Manager works on all Symmetrix platforms

- Unmapping causes hosts to lose access to the device
 - **unmap dev 13B:13F from dir 16A:0;**
- After unmapping the host operating system should be informed about the device's disappearance

The unmapping step causes the device(s) to no longer be presented to the front end port. Hosts accessing the devices should cease I/O to the device(s) before unmapping. It is important to perform a bus scan after unmapping so the host is made aware of the missing device.

Setting Port Characteristics

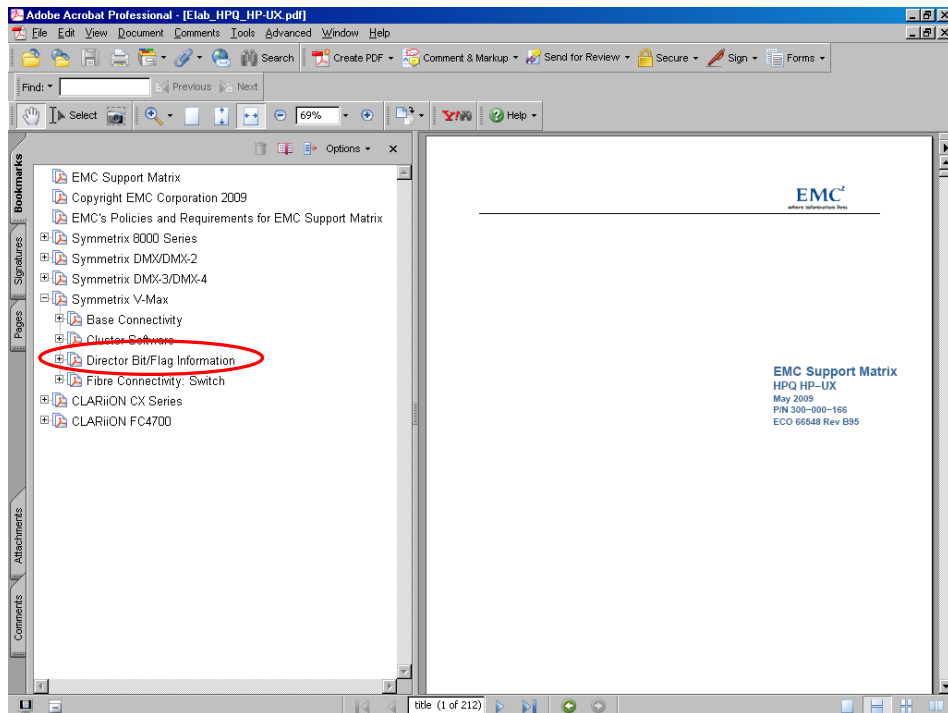
- Port characteristics are set on the Symmetrix director port
- Port can have no mapped devices
- Common settings include
 - Common_serial_Number
 - SCSI_3
 - SPC2_Protocol_Version
 - Volume_Set_Addresssing
- Port characteristics can be overridden by setting initiator flags using **symaccess**
- Important for 5874
 - ACLX flag allows storage autoprovisioning for the port

Setting front end port flags allows the FA port to be compatible with different types of hosts and fibre topologies. The Common Serial Number, SCSI3 and SPC2 Protocol version are used across a variety of platforms. Volume set addressing is used by HP-UX hosts.

Front end port flags can be overridden by the setting of HBA flags by using the **symaccess** command.

To use auto provisioning groups on Symmetrix V-Max the ACLX flag must be enabled on the port

O/S Specific Port Settings found in E-lab Navigator



© 2009 EMC Corporation. All rights reserved.

Device Creation and Mapping - 20

Mercer Road
EMC Education Services

E-lab navigator, which can be accessed through Powerlink, contains PDF copies of the support matrix for each operating system. In the section under Bit/Flag information you can find the port settings recommended for each operating system.

Settings for HP-UX

EMC Engenuity 5874						
No.	Operating System	Topology	Required Director Bit Setting	Optional Director Bit Setting	Storage Operating Environment	Storage Array
1	HPQ HP-UX 11i v3 (HP-UX 11.31)	FC-AL	(C) Common Serial Number; (SPC-2) SPC-2 compliance; (V) Enable Volume Set Addressing	(ACLX) Initiator Group; (OS2007) Host SCSI Compliance 2007 ¹	EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max
2	HPQ HP-UX 11i v3 (HP-UX 11.31)	FC-AL; FC-SW	(C) Common Serial Number; (V) Enable Volume Set Addressing		EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max
3	HPQ HP-UX 11i v3 (HP-UX 11.31)	FC-SW	(C) Common Serial Number; (PP) Enable Point-to-point; (SPC-2) SPC-2 compliance; (V) Enable Volume Set Addressing	(ACLX) Initiator Group; (OS2007) Host SCSI Compliance 2007 ¹	EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max
4	HPQ HP-UX 11i v1.0 (HP-UX 11.11), v2 (HP-UX 11.23)	FC-AL	(C) Common Serial Number; (FA LOOP ID) FA Loop ID; (V) Enable Volume Set Addressing	(ACLX) Initiator Group	EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max
5	HPQ HP-UX 11i v1.0 (HP-UX 11.11), v2 (HP-UX 11.23)	FC-AL; FC-SW	(C) Common Serial Number; (FA LOOP ID) FA Loop ID; (V) Enable Volume Set Addressing		EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max
6	HPQ HP-UX 11i v1.0 (HP-UX 11.11), v2 (HP-UX 11.23)	FC-SW	(C) Common Serial Number; (PP) Enable Point-to-point; (V) Enable Volume Set Addressing	(ACLX) Initiator Group	EMC Engenuity 5874: 121,102, 122,103	EMC Symmetrix V-Max

1. Recommended to enable os07 for HP-UX 11i v3 (HP-UX 11.31) initiators, refer to EMC HP-UX Host Connectivity Guide.

```
# symcfg list -fa 7E -p 0 -sid 80 -v
```

```
.....
Fibre Specific Flags
```

```
{
  Volume_Set_Addresssing(V)      : Enabled
  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)           : Disabled
}
```

© 2009 EMC Corporation. All rights reserved.

Device Creation and Mapping - 21

 Mercer Road
EMC Education Services

The example here shows an excerpt from the E-lab Navigator Support Matrix for HP-UX. The output of the symcfg command shows the port settings for the port to which the HP host is connected.

Device Reservations

- Devices can be marked as 'reserved' for future configuration or masking changes
 - The idea is to ensure that the device does not get into a state that would prevent the future configuration or masking change
- Reservations are used only by
 - symconfigure
 - symmask / symaccess
 - Symmetrix Management Console (SMC)
- Reservation file resides on the Symmetrix SFS device
- User creates a reservation and is given a reserve_id
 - The reserve_id must be supplied with the symmask or symconfigure commands to proceed with operations of Reserved devices
- Reservations can expire or be released
- There are two kinds of reservations: Advisory & Enforced

The configuration change utility can be used 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.

.

Enforced vs. Advisory Reservations

- **Enforced Reservations**
 - Enforced by the SYMAPI library, not the application.
 - The user must provide the reserve_id (through the application) if making any changes to reserved devices.
- **Advisory Reservations**
 - Enforced, by co-operating applications
 - symaccess, symmask (on DMX), symconfigure and SMC
- **Default values of the parameters in the options file**
 - SYMAPI_ENABLE_DEVICE_RESERVATIONS=TRUE
 - SYMAPI_ENFORCE_DEVICE_RESERVATIONS=TRUE

There are two types of device reservations:

1. **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. Applications developed before Solutions Enabler V6.4 may not be able to process reservation IDs.
2. **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 the user fails to explicitly do so.

Reservations can be Bypassed

- A masking or config change command can fail with the error:
 - A device or address in the request is reserved and this operation can not proceed
- User can
 - List the reservations
 - Respect the reservation and not attempt the change
 - Re-issue the command with the option
 - **symconfigure -sid 12 -file delete.cmd -reserve_id 5 commit**
 - Release the reservation with the command
 - **symconfigure -sid 3241 release -reserve_id 5,7**

If a user tries to apply a change to a reserved device, Enginuity will issue an error message. The user can either respect the reservation or bypass the reservation by re-issuing the command with the `reserve_id`.

Device Reservation Example -1

```
DMX8HP3/usr/sengupta> symconfigure -reserved list
```

```
Symmetrix ID: 000194900182
```

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
------------	---------------	----------	-------	---------	----------------

No device reservations were found

```
DMX8HP3/usr/sengupta> symdev show 29 -sid 82
```

```
Device Physical Name : /dev/rdisk/c24t0d1
```

```
Device Symmetrix Name : 0029
```

Front Director Paths (4):

PdevName	POWERPATH	DIRECTOR	PORT		LUN			
	Type	Type Num	Sts	VBUS	TID	SYMM	Host	
/dev/rdisk/c24t0d1	PARENT	FA	08E:0	RW	000	00	001	001
/dev/rdisk/c26t0d1	CHILD	FA	07E:0	RW	000	00	001	001
/dev/rdisk/c28t0d1	CHILD	FA	08E:0	RW	000	00	001	001
/dev/rdisk/c30t0d1	CHILD	FA	07E:0	RW	000	00	001	001

This is a simple example of how device reservations work. To start, Symmetrix 82 shows no device reservations. We then examine device 029, which is mapped to ports 7E:0 and 8E:0 on Symmetrix 82.

Device Reservation Example -2

```
DMX8HP3/usr/sengupta> symconfigure -sid 82 -cmd "reserve dev 029;" -owner Shekhar
  -comment "Testing Reservations" reserve -nop
```

A Device Reservation update is in progress.

Reserve_id = 000001

The Device Reservation update has succeeded.

```
DMX8HP3/usr/sengupta> symconfigure -reserved list
```

Symmetrix ID: 000194900182

SYMMETRIX DEVICE RESERVATIONS

Reserve ID	Date Reserved	Flags	Owner	Devices	Port Addresses
000001	06/04 15:00	E-	Shekhar	0029	-
Comment: Testing Reservations					

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

Here we reserve device 29 with a comment and an owner name. Both the comment field and the owner field are required by the syntax rules.

Device Reservation Example -3

```
DMX8HP3> symconfigure -sid 82 -cmd "map dev 029 to dir 7F:0 lun=041;" commit -nop
```

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

```
Establishing a configuration change session.....Established.
```

```
Processing symmetrix 000194900182
```

```
Device 0x0029 is reserved by reservation id = 000001
```

```
Error occurred while Checking Device Reservations
```

```
A device or address in the request is reserved and this operation can not proceed
```

```
Terminating the configuration change session.....Done.
```

```
The configuration change session has failed.
```

```
DMX8HP3> symconfigure -sid 82 release -reserve_id 1 -nop
```

```
A Device Reservation update is in progress.
```

```
The Device Reservation update has succeeded.
```

```
DMX8HP3/usr/sengupta> symconfigure -reserved list
```

```
Symmetrix ID: 000194900182
```

```
.....
```

```
No device reservations were found
```

An attempt to map device 029 now results in an error with a clear message as to why. We now have a choice to respect the reservation and leave the device alone. We can also release the reservation, after which the device is free to be mapped.

Lecture Assignment 2-1

- Review Table 3 on page 37 (Array Control Version 7)
 - How many mirror positions does a VDEV have
- Log into the lab host
 - Identify the serial number of the local Symmetrix
 - **symcfg list**
 - List virtual devices in the Symmetrix and examine one of them, e.g.
 - **symdev list -vdev -sid <SN of local Symmetrix>**
 - **symdev show 45 -sid <SN of local Symmetrix>**
 - How many mirrors does a VDEV have (Look for Mirror Set Type)
 - Does this information match what you see in the table?
 - If the local Symmetrix has diskless devices examine the mirror positions on those
 - **symdev list -dldev -sid <SN of local Symmetrix>**
 - **symdev show 95 -sid <SN of local Symmetrix>**



Module Summary

Key points covered in this module:

- Device Creation and Deletion
- Formation and Dissolution of Meta volumes
- Mapping and Unmapping Devices
- Setting Port Characteristics
- Setting Device Reservations

These are the key points covered in this module. Please take a moment to review them.

Module 3: Symmetrix and Device Attributes

Upon completion of this module, you will be able to:

- Set Device Attributes
- Create Dynamic RDF Groups and Pairs
- Set RDF Group Attributes
- Manage Snap Pools



The objectives for this module are shown here. Please take a moment to read them.

Device Attributes for Enginuity 5874

At Creation	Used by
ckd_meta	Mainframe applications. These are different from FBA meta volumes
savedev	TimeFinder Snap or DSE (Delta Set Extension) pools
datadev	Virtual provisioning for storing data written to thin devices
SCSI_3_persistent_reserv	Unix and Windows cluster software
ACLX	Used as a gatekeeper to the Auto provisioning database, which resides on the Symmetrix file system
dyn_rdf, dyn_rdf1, dyn_rdf2	Dynamic RDF

A Symmetrix device can have a number of attributes that are settable at device creation time. The attributes described here are documented in Chapter 1 of the Array Controls Guide.

CKD_META volumes are the equivalent of striped meta volumes in the Mainframe world.

Save devices provide the storage for TimeFinder Snap. When an application writes to a TF/Snap Virtual device, the data is stored on the save device. Save devices are also used as temporary storage to handle overflow data when an RDF/A delta set runs out of space in cache memory.

Datadevs are the repository for data written to Thin devices.

The SCSI_3 persistent reservation attribute, sometimes called the PER bit is used by a number of Unix cluster products such as Veritas and Sun.

The ACLX flag is set on a device, which acts as a gatekeeper to the auto provisioning information that resides on the Symmetrix file system. There is only one ACLX device per Symmetrix. In addition, ports have to have the ACLX flag enabled to participate in autoprovisioning.

The dynamic RDF attributes allow a device to be configured as a dynamic R1 only (dyn_rdf1), dynamic R2 only (dyn_rdf2), or dynamic RDF1 or RDF2 device (dyn_rdf). Except under special circumstances, most devices are assigned the dyn_rdf flag

Device Attributes for Enginuity 5874

After Creation	Action
SCSI_3_persi st_reserv	Unix and Windows cluster software
ACLX	Used as a gatekeeper to the Auto provisioning database, which resides on the Symmetrix file system
dyn_rdf, dyn_rdf1, dyn_rdf2	Dynamic RDF

Other attributes are supported in previous versions of Enginuity. Check the Array Controls Guide for more information

The attributes shown here are ones that can be assigned to a device after device creation. There are a number of other attributes that can be set on DMX devices but those have been discontinued in 5874

Dynamic RDF Groups

- Dynamic RDF groups allow creation and deletion of SRDF groups using the `symrdf` command
- Works over switched SRDF network
- Sample command:

```
symrdf addgrp -label <label> -sid <xx> -rdfig <m> -
  dir <i> -remote_sid <yy> -remote_rdfg <n> -
  remote_dir <j>
```
- Additional documentation is located in Chapters 3 and 7 of the SYMCLI SRDF manual



Dynamic RDF groups can be created between two Symmetrix arrays that are zoned together through a fiber or Gig-e switch. Creation and removal of the groups can be done quickly through the use of the `symrdf` command and do not require intervention from an EMC customer services engineer.

Notes on the demo:

- a) View all arrays visible from the host. Symmetrix 82 is local, 81 is remote
- b) Examine the RDF directors on both Symmetrix arrays. Note that director 7H is empty on both arrays and that RDF group number 64 is the highest group on both
- c) The syntax on this slide to create an RDF group numbered 100 and label it “mygroup”
- d) We then examine director 7H on both Symmetrix arrays to verify that group 100 has been created
- e) We then use the command “`symcfg list -rdfig 100`” to verify that the label is “mygroup”

Dynamic RDF Devices

- Use the **symrdf** command to permit quick creation and deletion of RDF pairs
- Devices can be:
 - R1 capable
 - R2 capable
 - R1 and R2 capable
- Dynamic RDF attribute of a device can be examined in the output of **symdev show**:

```
# symdev show 95 -sid 35
● ● ●
Dynamic RDF Capability    : RDF1_OR_RDF2_Capable
● ● ●
```

Dynamic RDF devices can only exist in a Symmetrix that has the Dynamic RDF feature enabled. They can be created to be RDF1 capable, RDF2 capable, or RDF1 or RDF2 capable (as shown above).

Dynamic RDF Pairs

- The Symmetrix must be set to allow Dynamic RDF Pairs
Dynamic RDF Configuration State : Enabled

- A device file containing device pairs comprising source and target devices is needed



- Sample command:

```
symrdf createpair -file <device file> -sid <xx> -  
rdfg <n> -type RDF1 -establish
```

- Additional documentation is located in Chapters 2 and 7 of the SYMCLI SRDF manual

Dynamic RDF pairs can be created or deleted using the “symrdf” command. They can belong to static or dynamic RDF groups.

Notes on the demo:

- a) We log in to host sun3 and issue a `symcfg list` to confirm that the local Symmetrix id is 82 and the remote Symmetrix id is 81
- b) Examine devices B3 and B4 on the local array. They are 2-Way-Mir
- c) `symdev show` command on B3 and B4 confirm they’re dynamic RDF capable
- d) We then examine devices B3 and B4 on Symmetrix 81 the remote array. They too are dynamic RDF capable
- e) We create a file called `rdfpairs` in the local directory and provide the dynamic RDF pairing relationships that we intend to create
- f) Next we run the `createpair` command as shown on this slide
- g) Finally we run an RDF query on those two devices to verify that the pair creation succeeded

RDF Group Attributes for SRDF/A

At Creation	Used by
session_prio rity	SRDF/A sessions can be assigned a priority from 1-64. Lower numbers have higher priority.
minimum_cycl e_time	Minimum time to wait before attempting a cycle switch
rdfa_transmi t_idle	Enabled by default this feature can help SRDF/A override temporary network problems
rdf_dse_pool	The name of a collection of save devices used for DSE
rdfa_dse_aut ostart	Specifies whether SRDF/A DSE is automatically activated when SRDF/A is enabled. Disabled by default
rdf_dse_thre shold	Percentage of WP limit after which data will spill over to disk

The attributes assignable to RDF/A groups are shown here. Session priority is used when there are multiple RDF/A groups being used in a Symmetrix. In the event of cache shortage, the lower priority session (higher number) will drop before a higher priority session is dropped.

The minimum cycle timer measures minimum the time that has to elapse before a cycle switch is undertaken.

Transmit idle is a default setting of all RDF groups. When Transmit Idle is enabled on an RDF/A group, a temporary loss of network connectivity between source and target can be overcome by storing the writes in cache as long as there is cache available. If network connectivity returns before cache overflows, SRDF/A can continue cycle switching. If cache overflows before the network is restored, the session is dropped.

The DSE (Delta Set Extension) related attributes refer to the devices that are used for storing overflow data from an SRDF/A cycle when the cache overflows.

SRDF/A Session Characteristics

- Session priority
 - Sessions can be assigned a value between 1 and 64
 - If multiple SRDF/A sessions are contending for cache and there is no more cache available, the lower priority session drops first
- Minimum Cycle Time
 - Minimum amount of time after which SRDF/A will attempt to switch cycles (aka delta sets)
 - Value of cycle timer can be set from 1 – 59 secs.



Notes on the demo:

We examine the group we created earlier using the command

```
symcfg list -rdfg 100 -rdfa
```

and note that Transmit Idle is enabled, the cycle timer is 30 seconds and the session priority is 33,

A configuration manager command file called `rdfaparms` containing the lines

```
set rdf group 100 minimum_cycle_time=1;
set rdf group 100 rdfa_transmit_idle=Disable;
set rdf group 100 session_priority=20;
```

is executed with `symconfigure` and the changes are made. A second examination of the RDF group displays the effect of the changes that were just made.

Managing Device Pools

- Three kinds of pools
 - snap (contains save devices)
 - rdfa_dse (contains save devices)
 - thin (contains data devices – covered under Virtual Provisioning)
- When save devices are created, they are assigned to a default pool and are available for use by TimeFinder/Snap
- When pools are created they have to be named and identified as TF/Snap or RDF/A DSE pools
- Save devices can be moved into named pools
- Maximum number of pools in a Symmetrix is 510

There are three kinds of device pools supported by Enginuity. Snap pools and DSE pools contain save devices. These devices can be used in either kind of pool. Thin pools contain data devices, which cannot be used in Snap or RDF DSE pools.

A Save device is assigned to a default pool after creation. These can be used by TimeFinder/Snap but not RDF/DSE. If both RDF and Snap pools are in use in the Symmetrix, it is a good practice to place devices into named pools that have been designated as Snap or DSE pools.

The maximum number of pools a Symmetrix can support is 510.

SRDF/A DSE Pools

- Can be shared with multiple SRDF/A groups
- A single SRDF/A group can have at most one of each kind of pool associated with it
 - FBA
 - CKD 3380
 - CKD 3390
 - AS400
- A DSE pool can contain only one kind of the above four kinds of devices

Since SRDF devices can belong to a variety of operating systems with different emulations, the DSE pools used with SRDF can contain any one of four kinds of devices. Each DSE pool can hold only one kind of device.

TimeFinder/Snap Save Pools

- All save devices initially belong to a DEFAULT_POOL
- The DEFAULT_POOL is available for use by TimeFinder/Snap
- It is also possible to create named TimeFinder/Snap pools and move save devices into them
- To use a named snap pool the pool name can be specified when a TF/Snap session is created
- Pool space can be monitored using SYMCLI, SMC or Event daemon

When a save device gets created, it is initially placed into a default pool. The devices in the default pool are available for use by TimeFinder/Snap but not RDF/DSE. If both Snap and RDF save pools are being used in a Symmetrix, it is best to create named pools that are explicitly designated for use by TimeFinder/Snap or RDF/DSE.

Lecture Assignment 3-1

- Refer to pages 91 and 92 of the Array Control Guide
- Note the correct syntax for
 - Creating a pool
 - Enabling and disabling devices in the pool
 - Adding and removing devices from the pool



Notes on demo:

- a) We check the save devices that exist on the Symmetrix. There are 16 from 17D to 16C
- b) List the snap pools and we notice that the default pool is enabled
- c) Using `symsnap show` we see that all 16 save devices are included in the list and the pool is enabled
- d) We then create a pool called mypool
- e) We disable 17D and 17E in preparation for moving them to mypool
- f) We add 17D and 17E to mypool and enable the devices
- g) We issue a `show` command on mypool and note that the pool is now enabled with devices 17D and 17E participating in the pool

Module Summary

Key points covered in this module:

- Device Attributes
- Dynamic RDF Groups and Pairs
- RDF Group Attributes
- Snap Pools

These are the key points covered in this module. Please take a moment to review them.

Module 4: Autoprovisioning Groups

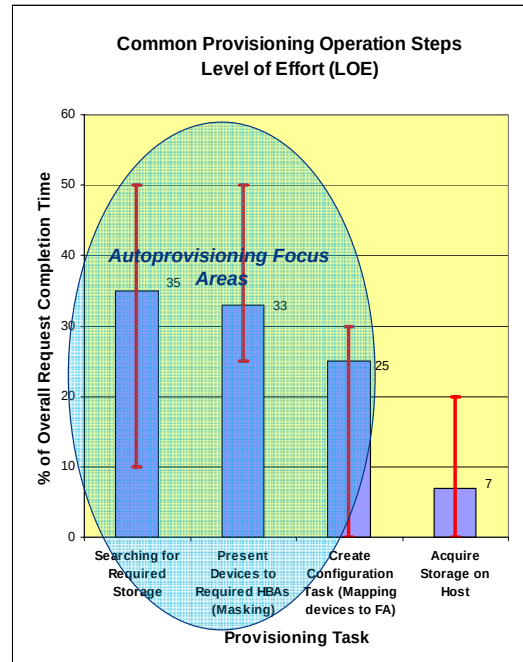
Upon completion of this module, you will be able to:

- Describe Auto-provisioning Groups
- List the steps to mask Devices on Symmetrix-VMax hardware
- Identify HBA flags that can be set with the symaccess command
- Describe automatic mapping and optional unmapping devices with autoprovisioning

The objectives for this module are shown here. Please take a moment to read them.

Background

- Research has shown that in during storage provisioning, the majority of time is spent
 - Searching for required storage on arrays
 - Mapping and masking devices which require a separate task for each initiator / port combination
- *“It’s too complex and takes too long” – Customer Complaint*
- Auto-provisioning Groups
 - Focuses ease-of-use improvements on the most time consuming day-to-day provisioning tasks



© 2009 EMC Corporation. All rights reserved.

Autoprovisioning Groups - 2

 Mercer Road
 EMC Education Services

Symmetrix Autoprovisioning represents a fundamentally new way of provisioning Symmetrix storage . It makes the task of Storage administrators easier by reducing the number of steps that are needed to provision storage

Search facilitated by SYMCLI Device Filtering

- Options in “symdev list” and SMC facilitate filtering
 - N
 - Sets the number of devices to list. The # specifies the maximum number of devices to return. The actual number returned may be less than the specified number if fewer devices exist. The default is to list all devices.
 - RAID1, RAID5 [-protection <3+1 | 7+1>], RAID6 [-protection <6+2 | 14+2>]
 - Lists RAID-1, RAID-5 and RAID-6 devices
 - CAP
 - Sets the device capacity to a specific value (in megabytes) for the selection criteria to be listed.
- To find 5 unmapped standard 8.6 GB RAID5 (7+1) devices in disk group 1

```
symdev list -sid 1201 -raid5 -protection 7+1 -noport -N 5  
-nobcv -disk_group 1 -cap 8631
```

Utilizing device filtering flags will help to expedite the storage selection process by allowing you to define, in advance, the characteristics of the devices you want to display in the output of “symdev list” statements

Filtering mechanisms can work well within scripts. For example: Perform above search and add found devices into an existing device group called “appgrp2”.

```
for dev in $(symdev list -sid 1201 -raid5 -protection 7+1 -noport -N 5 -nobcv -  
disk_group 1 -cap 8631 | grep ????:? | awk '{print $1}'); do symld -g appgrp2 add dev  
$dev -sid 1201; done
```

Rationale for Auto-provisioning

- Larger and larger arrays containing more and more devices
- Other vendors including Clariion use a similar style of masking
- Performed through use of the **symaccess** command

As the number of volumes in a single array continues to climb higher, a more flexible scheme for provisioning storage needed to be developed. Other vendors such as Clariion already use this kind of provisioning. Autoprovisioning in the Symmetrix V-Max is achieved through the use of the **symaccess** command.

Advantages of Autoprovisioning

- Initiators can be dynamically added or removed from initiator groups
- Ports can be dynamically added or removed from port groups
- Storage can be dynamically added or removed from storage groups

In storage terminology a SCSI initiator is the endpoint that initiates a SCSI session, that is, sends a SCSI command. Thus host adapters are initiators.

On the other hand, a SCSI target is the endpoint that does *not* initiate sessions, but instead waits for initiators' commands and provides required input/output data transfers. The target usually provides to the initiators one or more LUNs, because otherwise no read or write command would be possible.

Auto provisioning represents a change from the way Symmetrix devices used to be masked. It is an easier, faster way to handle provisioning once the groups have been created. It allows you to mask multiple devices, ports and HBAs by placing them into groups. These groups can be dynamically altered to give the host access to new storage.

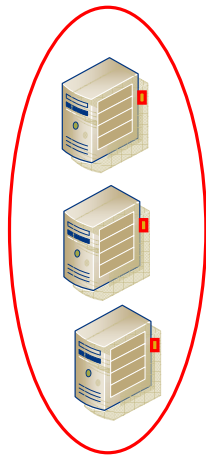
There is no device masking database. There is no initialization, backup, or restore. With the symaccess command, all groups and views are backed up to a file, and can be restored from a backup file.

Volume visibility and the LUN offset on an initiator are no longer supported.

Heterogeneous host types are no longer supported.

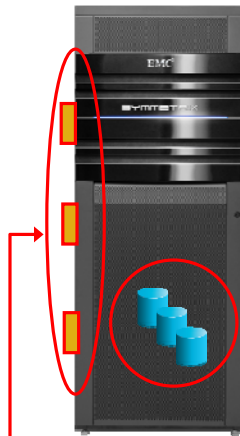
The HBA flags option is supported.

Auto-provisioning Groups



Initiator Group

Contains HBA WWNs



Port Group

Contains Port Names



Storage Group

Contains Device Names

Group Names

- Up to 64 characters long
- Case insensitive
- Unique per group type

Auto provisioning requires the use of Initiator Groups, Port Groups and Storage Groups.

Initiator groups contains host initiator or iSCSI names. Port Groups contain valid front end FA or Gig-E ports. Storage groups contain Symmetrix devices.

Symaccess Operations

Control	Action
create	Creates a view or group of the specified type
delete	Deletes a view or a group
add	Adds elements to a specified group
remove	Removes elements from a specified group
enable	Enables security information on a port
disable	Disables security information on a port
set	Sets initiator attributes
discover	Discovers the WWN of the HBAs
backup	Creates a file containing all the group and view information
restore	Restores group, view and security information from a file

symaccess is a function rich command and has a number of control actions and a couple of display actions. A summary of the control actions is provided here.



Symaccess Operations - 2

Display	Action
list	Lists group or view information
show	Shows detailed information on groups and views

list and show are the two display actions available with symmigrate.

Storage Groups

- Contain Symmetrix Devices
- Device reservations are enforced when adding devices to a storage group

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 will be enforced whenever devices are being added to a storage group.

Storage Group Command Syntax

```
#symaccess create -sid 458 -name SG_1 -type storage -devs 50:52
```



Symmetrix Devices 50, 51 and 52

You can create a storage group using a range of devices, device names, device group devices, or a device file. The symaccess syntax for creating a storage group is:

```
symaccess -sid SymmID create -name GroupName -type storage
```

```
[ < [-reserve_id ResvID [, ResvID [, ResvID ...]]]
```

```
devs SymDevStart:SymDevEnd | SymDevName,  
SymDevName,SymDevName... > |
```

```
<-g DgName [-std] [-bcv] [-vdev] [-tgt] > |
```

```
< -file DeviceFileName [src] [tgt] > ]
```

Common Operations on Storage Groups

- Create the storage group
 - Optionally populate with device range, device group or a device file
 - `symaccess create -sid 80 -name sun1_StorageGroup -type storage devs 0B1:0CC`
- Add devices to Storage group
 - `symaccess -sid 80 -name sun1_StorageGroup -type storage add devs CD:F4`
- Remove devices from Storage group
 - Optionally, devices can be unmapped when they are removed from a storage group that is part of a view
 - `symaccess -sid 80 -name sun1_StorageGroup -type storage -unmap remove devs CD:F4`
- List group contents in varying amount of detail
 - `symaccess list -type storage -name sun1_StorageGroup -sid 80`
 - `symaccess show sun1_StorageGroup -type storage -sid 80`
- The storage group can be renamed or deleted
 - `symaccess -sid 80 rename -name sun1_StorageGroup -new_name GHI -type storage`
 - `symaccess -sid 80 -name sun1_StorageGroup -type storage delete`
- List the group that a device belongs to
 - `symaccess list -sid 80 -type storage -dev B5`

These are some of the operations that can be performed on the storage group.

Port Groups

- Contain valid front end ports
- A port can belong to more than one port group
- Only Fibre and Gig-E ports on front end directors allowed
- Ports must have ACLX flag enabled

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.

Port Group Creation Command Syntax

V-Max Front end Ports

7E:0

7G:1

8F:0



```
symaccess create -sid 80 -name PG_1 -type port -dirport 7E:0,7G:1,8F:0
```

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. The syntax for port group creation is:

```
symaccess -sid SymmID create -name GroupName -type port [-  
dirport Dir:Port [,Dir:Port...]]
```

Common Operations on Port Groups

- Create the port group
 - `symaccess create -sid 80 -name lin1_Ports -type port`
- Add ports to groups
 - `symaccess -sid 80 -name lin1_Ports -type port -dirport 7F:1 add`
- Remove ports from Port groups.
 - Optionally, devices can be unmapped from the removed ports, if the port group is part of a view
 - `symaccess -sid 80 -name lin1_Ports -type port -dirport 7F:1 remove -unmap`
- List group contents in varying amount of detail
 - `symaccess list -name lin1_Ports -sid 80`
 - `symaccess show lin1_Ports -type port -sid 80`
- Rename or delete group
 - `symaccess -sid 80 rename -name lin1_Ports -new_name DEF -type port`
 - `symaccess -sid 80 -name lin1_StorageGroup -type storage delete`
- List the group(s) that a port belongs to
 - `symaccess list -sid 80 -type port -dirport 8F:1`

These are some of the operations that can be performed on a port group.

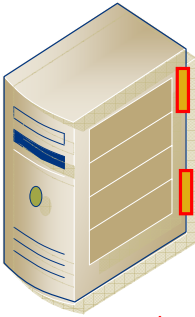
Initiator Group

- Contains Fibre WWNs or iSCSI names
- Maximum of 32 entries
- An initiators may belong to only one IG
- IGs can be cascaded one deep, an IG can belong to one or more IGs
- Example:
 - Initiator Group IG_1 contains WWN1
 - Initiator Group IG_2 contains WWN2
 - Initiator Group IG_Both can contain IG_1 and IG_2

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.

Initiator Group Creation Syntax



Initiator WWNs

50060b00000788a8

50060b0000077fbc

```
#symaccess create -sid 80 -name hp1_Initiators -type initiator
#symaccess -sid 80 -name hp1_Initiators -type initiator -wwn 50060b00000788a8 add
#symaccess -sid 80 -name hp1_Initiators -type initiator -wwn 50060b0000077fbc add
```

OR

```
#symaccess create -sid 80 -name hp1_Initiators -type initiator -file HBA_WWNS
```

File HBA_WWNS contains

wwn:50060b00000788a8

wwn:50060b0000077fbc

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 is:

```
symaccess -sid SymmID
```

```
create -name GroupName -type initiator [ -wwn wwn | -iscsi iscsi
-file InitiatorFilename | -ig InitiatorGroupName ]
```

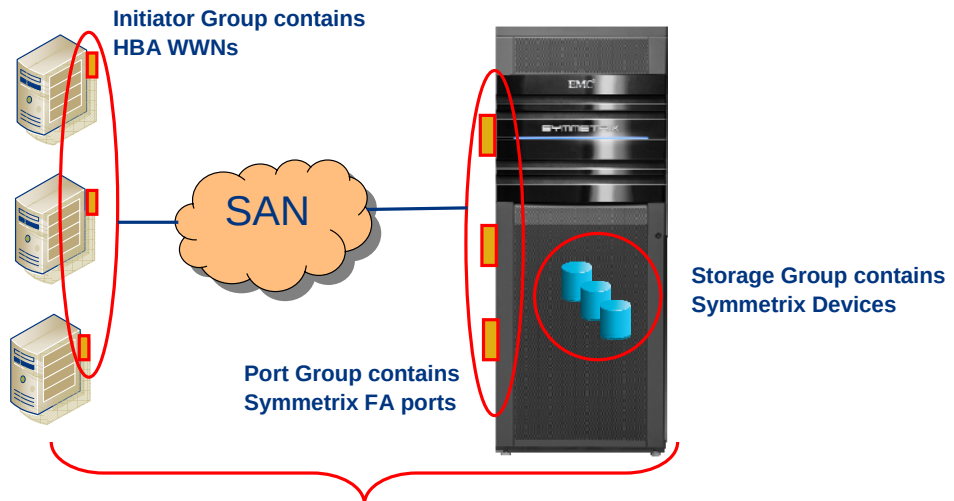

Common Operations for Initiator Groups

- Create initiator group
 - `symaccess create -sid 80 -name sun1_Initiators -type initiator`
- Add initiator to group
 - `symaccess -sid 80 -name sun1_Initiators -type initiator -wwn 10000000c92ab6de add`
- Remove initiator from initiator group
 - `symaccess -sid 80 -name sun1_Initiators -type initiator -wwn 10000000c92ab6de remove`
- List group contents in varying amount of detail
 - `symaccess list -name sun1_initiators -sid 80`
 - `symaccess show sun1_initiators -type initiator -sid 80`
 - `symaccess show sun1_initiators -type initiator -sid 80 -detail`
- Rename or delete group
 - `symaccess -sid 80 rename -name sun1_Initiators -new_name ABC -type initiator`
 - `symaccess -sid 80 -name sun1_Initiators -type initiator delete`
- List the group(s) that an initiator belongs to
 - `symaccess list -sid 80 -type initiator -wwn 10000000c93124ae`

Here are some examples of using Initiator groups

Masking View Command Syntax

```
#symaccess create view -sid 458 -name MV_1 -sg SG_1 -pg PG_1
```



A Masking View Contains: 1 Initiator Group + 1 Port Group + 1 Storage Group

© 2009 EMC Corporation. All rights reserved.

Autoprovisioning Groups - 18  Mercer Road
EMC Education Services

On a Symmetrix V-Max autoprovisioning 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.

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 syntax is:

```
symaccess -sid SymmID create view -name ViewName -sg  
StorageGroupName
```

```
-pg PortGroupName -ig InitiatorGroupName
```

```
[ < [-reserve_id ResvID [, ResvID [, ResvID ...]]]
```

```
[-lun Addr]
```

Common Operations on Views

- View creation requires populated storage, port and initiator groups
 - Conversely, you cannot empty out a group that belongs to a view
 - `symaccess create -sid 80 view -name ibm1_MaskingView -sg ibm1_StorageGroup -pg ibm1_Ports -ig ibm1_Initiators`
- Views can be deleted and renamed
 - `symaccess -sid 80 -name ibm1_MaskingView delete view`
 - `symaccess -sid 80 rename view -name ibm1_MaskingView -new_name XYZ`
- Display contents of views in varying amounts of detail
 - `symaccess list view -name WIN1_MaskingView -sid 80`
 - `symaccess list view -name WIN1_MaskingView -sid 80 -v`
 - `symaccess show view WIN1_MaskingView -sid 80`
- Views can be backed up so they can be restored if needed
 - `symaccess -f WIN1_MaskingView -sid 80 backup`
 - `symaccess -f WIN1_MaskingView -sid 80 restor`

These are some of the operations that can be performed on masking views

Reason for Dynamic LUN Addressing

- Inside the Symmetrix each FA assigns a device a LUN value after masking

```
# symcfg list -sid 80 -avail -addr -fa 7F -p 0
```

Director			Device Name		Attr	Address		
Ident	Symbolic	Port	Sym	Physical		VBUS	TID	LUN
FA-7F	07F	0	0028	c2t50000972C002D158d0s*	ACLX	0	00	000
			00B1	/dev/rdisk/emcpower50c		0	00	001
			00B2	/dev/rdisk/emcpower54c		0	00	002
			00B3	/dev/rdisk/emcpower58c		0	00	003
			0194	Not Visible		0	00	044
			-	AVAILABLE		0	00	045 *

- Sometimes the LUN values assigned by the FA are unsuitable for the host
 - FA assigned values can exceed the HBA's ability to address
 - Applications may require a specific LUN number to function correctly

A Symmetrix FA port is capable of supporting 4096 mapped devices. It assigns LUN numbers to mapped devices starting at 0 and counting up in 3 hexadecimal digits. For some host environments this is a problem, because some host HBAs are limited in the highest LUN that they can support. In other instances, applications might rely on a certain LUN such as LUN 0. Dynamic LUN addressing addresses this problem.

Dynamic LUN Addressing used with Autoprovisioning

- Dynamic LUN Addressing is enabled by default.
- By default Symmetrix array assigns the next available LUN address on the FA port when the masking view is created.
- If needed user can define LUN address making LUN addresses consistent across FAs
- Example: Create a masking view named Prod1_View utilizing pre-existing group components Prod1_IG, Prod1_PG, and Prod1_SG. Optional - use a starting LUN address of 040 for devices

```
symaccess -sid 1201 create view -name Prod1_View -sg Prod1_SG -pg  
Prod1_PG -ig Prod1_IG -lun 040
```

Dynamic LUN addressing allows specific LUN values to be assigned, either manually or automatically, to each Symmetrix device that is being masked to an HBA, regardless of what LUN was assigned when the device was mapped to the FA. This eliminates the potential impact of the 256 LUNs per target limit of many HBAs by allowing LUN addresses between 0 and 255 to be specified on a per HBA World Wide Name basis. It also allows any device to be addressed as LUN 0 if a host requires that a device be assigned that LUN value.

Examples of List Output

```
DMX800SUN1/usr/sengupta> symaccess list view -sid 80
```

```
Symmetrix ID : 000194900180
```

Masking View Name	Initiator Group	Port Group	Storage Group
WIN1_MaskingView	WIN1_Initiators	WIN1_Ports	WIN1_StorageGroup
sun1_MaskingView	sun1_Initiators	sun1_Ports	sun1_StorageGroup
ibm1_MaskingView	ibm1_Initiators	ibm1_Ports	ibm1_StorageGroup
lin1_MaskingView	lin1_Initiators	lin1_Ports	lin1_StorageGroup
hp1_MaskingView	hp1_Initiators	hp1_Ports	hp1_StorageGroup

```
DMX800SUN1/usr/sengupta> symaccess list -type port -sid 80 -detail
```

```
Symmetrix ID : 000194900180
```

Port Group Name	Port Count	View Count
WIN1_Ports	2	1
hp1_Ports	2	1
ibm1_Ports	2	1
lin1_Ports	2	1
sun1_Ports	2	1

Here are two examples of how the list action works with symaccess

Examples of Show Output

```
DMX800SUN1/usr/sengupta> symaccess show hp1_StorageGroup -type storage -sid 80
```

```
Symmetrix ID      : 000194900180
Last updated at   : 04:07:26 PM on Tue May 19,2009
Storage Group Name : hp1_StorageGroup
Devices          : 0029:005D 005F 0061 0063 0065:006C
Masking View Names
{
  hp1_MaskingView
}
```

```
DMX800SUN1/usr/sengupta> symaccess show lin1_Initiators -type initiator -sid 80
```

```
Symmetrix ID      : 000194900180
Last updated at   : 04:07:21 PM on Tue May 19,2009
Initiator Group Name : lin1_Initiators
Host Initiators
{
  WWN :10000000c94eadda
  WWN :10000000c94eaddb
}
Masking View Names
{
  lin1_MaskingView
}
Parent Initiator Groups
{
  None
}
```

Here are two examples of how the show action works with symaccess

Setting HBA Flags

- The following HBA flags can be set on a per initiator basis

– Common_Serial_Number	[C]
– Disable_Q_Reset_on_UA	[D]
– Environ_Set	[E]
– Siemens	[S] *
– Volume_Set_Addresssing	[V]
– Avoid_Reset_Broadcast	[ARB]
– AS400	[AS4]
– OpenVMS	[OVMS]
– SCSI_3	[SC3]
– Sunapee	[SCL] *
– Sequent	[SEQ] *
– SPC2_Protocol_Version	[SPC2]
– SCSI_Support1	[OS2007]

Note:

Siemens, Sunapee and Sequent flags are only supported on DMX Arrays not on V-Max

Symmetrix arrays running Solutions Enabler V7.0 and Enginuity 5874, only allow you to set the HBA port flags on a per initiator basis. This feature allows specific host flags to be enabled and disabled on the director port.

Setting HBA port flags replaces setting the heterogeneous host configuration flags. 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.

Mapping and Unmapping with symaccess

- Mapping

- if **symaccess** finds unmapped devices during view creation, it will map the unmapped devices to all ports in the port group associated with the view
- In prior versions of Solutions Enabler, mapping of devices using **symconfigure** was an essential step before masking

- Unmapping

- The user has the option of
 - Unmapping all devices when the view is deleted
 - Unmapping devices that are removed from a storage group while that storage group is part of a view
 - Unmapping devices from the port if that port is removed from a port group (the port group has to be part of a view for this to happen)
- All affected devices including those mapped with **symconfigure** will be unmapped.

The **symaccess** command will map devices to a port if needed at the time of view creation. The mapping happens automatically without user intervention. However, it takes longer to create a masking view, if the devices have to be mapped as well.

Unmapping of devices can be performed when a view is deleted

```
symaccess -sid 80 delete view -name MV -unmap
```

Unmapping of devices that are part of a storage group which is participating in a masking view

```
symaccess -sid 80 -name SV -type storage remove dev 011 -unmap
```

Unmapping of devices from a port that is participating in a masking view when the port is removed from the masking view

```
symaccess -sid 80 -name PV -type port -dirport 7E:0 -unmap  
remove
```

Example of a Shared Storage Implementation



Solution

4 Initiator Groups	1 Port Group	4 Storage Groups	4 Masking Views
IG1 contains HBA1 WWN	PG contains port P1	SG1 contains GK1, GK2	MV1 holds IG1, PG, SG1
IG2 contains HBA2 WWN	PG contains port P1	SG2 contains GK3, GK4	MV2 holds IG2, PG, SG2
IG3 contains HBA3 WWN	PG contains port P1	SG3 contains GK5, GK6	MV3 holds IG3, PG, SG3
IG4 contains IG1, IG2, IG3	PG contains port P1	SG4 contains D1:D100	MV4 holds IG4, PG, SG4

© 2009 EMC Corporation. All rights reserved.

Autoprovisioning Groups - 26 Mercer Road
EMC Education Services

This is a practical example of how one might configure shared storage intended for a cluster or a shared database. In a clustered environment, some devices need to be seen by all the hosts in the cluster. Other devices such as gatekeepers may need to be seen only by individual hosts in the cluster.

To achieve this, FOUR initiator groups, FOUR storage groups, and FOUR masking views are created. We'll assume one port in one port group to keep the example simple.

The storage groups are straightforward. Each of the four pools of storage are placed in a storage group.

The first three initiator groups contain one WWN each. The fourth initiator group is a cascaded initiator group which contains the names of the three initiator groups, which contain the HBA WWNs. Using the cascaded initiator group it is possible to give all the initiator groups access to the shared storage while each individual initiator retains private access to its gatekeepers.

Using the 4 storage groups and 4 initiator groups it is now possible to construct 4 masking views

Steps to Replace an HBA

- The replace action can be used to allow a new HBA to take over the devices visible to the old HBA
- View old HBA WWN
 - **symaccess list logins**
- Swap out the old HBA board with the new HBA
- Discover the WWN of the new HBA
 - **symaccess discover hba or symaccess list hba**
- Use the replace action
 - **symaccess -sid 80 replace -wwn <WWN_old> -new_wwn <WWN_new>**
- Use the rename action to establish the new alias for the HBA
 - **symaccess discover hba -rename**

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:

```
symaccess replace -wwn wwn -new_wwn NewWWN [-noprompt]
```

Lecture Discussion Assignment 4-1

- Read the table (Table 12) on SCSI protocol flags in Chapter 1 of the Array Control Guide page 113
- Read the explanation for Volume Set addressing at the end of the table
- Read the section on Setting HBA Flags in Chapter 2 of the Array control Guide page 134-135
- Then discuss the following:
 - An HP host and a Solaris host are sharing the same FA port. Assuming that all other flags are the same between HP and Sun, how would you have the HP use Volume Set Addressing while the Sun does not?

Module Summary

Key points covered in this module:

- Auto-provisioning Groups
- Steps to mask Devices on Symmetrix-VMax hardware
- HBA flags that can be set with the symaccess command
- Automatic mapping and optional unmapping devices with autoprovisioning

These are the key points covered in this module. Please take a moment to review them.

Module 5: Virtual LUN Technology

Upon completion of this module, you will be able to:

- Describe Virtual LUN migration benefits
- List requirements for Virtual LUN migration
- Explain Virtual LUN migration operational flow
- Explain host considerations for migrating volumes
- Use the symmigrate command to perform a migration

The objectives for this module are shown here. Please take a moment to read them.

Reasons for for V-LUN Migration

- Migration of applications between tiers of storage
- Migration of applications from one class of RAID protection to another
- Information Lifecycle Management
- Performance Optimization

Virtual LUN Migration allows Symmetrix users to seamlessly move application data between different storage tiers. If data for an application is initially placed on a lower tier of storage and performance needs increase, V-LUN Migration allows the application to be moved to faster disks without interruption of production.

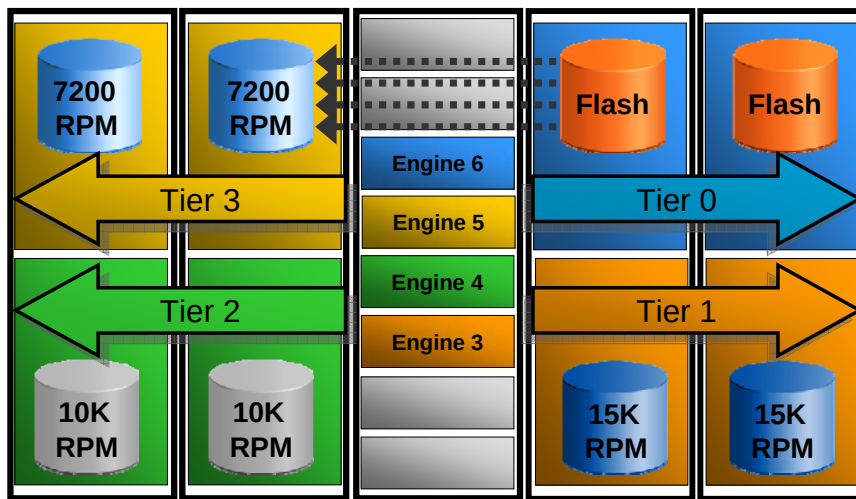
V-LUN migration permits applications to move from one class of RAID protection to another to save storage space, e.g. RAID-1 to RAID-5.

As data ages and the need for its availability diminishes, it can be moved to less expensive storage as part of the ILM process.

Performance optimization is another function of V-LUN migration as applications are moved between storage tiers

Enhanced Virtual LUN Technology

- Non-disruptive migration of devices and meta devices
 - Between RAID levels or Disk Groups
 - Key enabler for **storage tiering** within the array



© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 3

Mercer Road
EMC Education Services

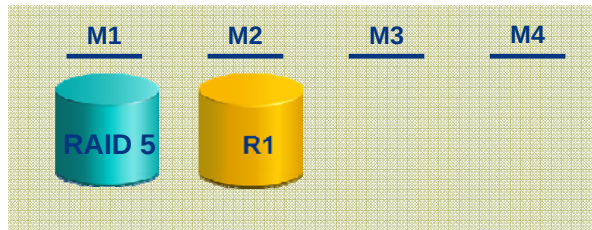
The Symmetrix V-Max Series with Enginuity 5874 enables users to perform non-disruptive migration of volumes among storage tiers within the same array and between RAID protection schemes. Enhanced Virtual LUN technology is supported for both open systems (FBA) and mainframe volumes (CKD), and includes support for meta volumes. Virtual LUN Technology is a licensed feature bundled under the Symmetrix Optimization offering which includes Symmetrix Optimizer. However, DRV volumes are not needed for Enhanced Virtual LUN Technology.

Virtual LUN technology enables data migration within an array without host or application disruption allowing an Information Lifecycle Management (ILM) strategy to be adapted over time by easily moving information throughout the storage system as requirements change. It can assist in system reconfiguration, performance improvement, and consolidation efforts all while helping maintain vital service levels.

In this example, data may start out on a set of tier 0 volumes such as Enterprise Flash Drives. Over time, that data may no longer warrant the high performance of Flash Drives. Migrating that data off array is time consuming and disruptive. However, leveraging Virtual LUN technology, this data can be moved to another tier in the same array without needing to shut down applications or systems and without disrupting local or remote replication.

RAID Virtual Architecture

- New RAID Architecture with Enginuity 5874
 - Abstracts RAID protection from Device
 - RAID Group occupies single mirror position
 - Enabling technology for Virtual LUN migration
 - Does NOT introduce new RAID type

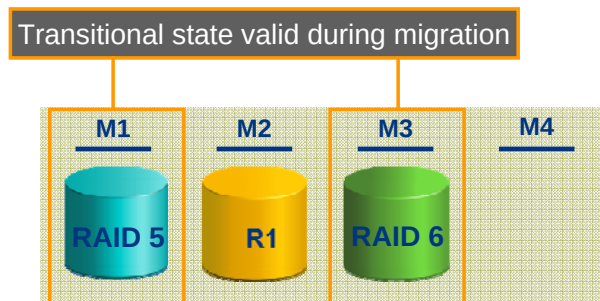


RAID Virtual Architecture implemented with Enginuity Operating Environment 5874 is the new method for handling device mirror position usage. RAID Virtual Architecture is an enabling technology for the Enhanced Virtual LUN technology. Note that RAID Virtual Architecture does not introduce any new RAID types. All device protection types use the same I/O execution engine which separates the mirror position interface from RAID internal operations.

With the RAID virtual architecture in Enginuity 5874 a mirror position holds a logical representation of a RAID group rather than a device, resulting in additional free mirror positions as demonstrated in this example. Notice the RAID 5 volume with SRDF protection consumes only two mirror positions. The RAID 5 group occupies only one mirror position with the SRDF protection occupying a second position. This frees two mirror positions for other operations such as a migration to another RAID type.

RVA Operations and Limitations

- 2 RAID Groups Attached
 - During migration only
 - RAID Groups can be in different Disk Groups
 - Non-disruptive to local and remote replication

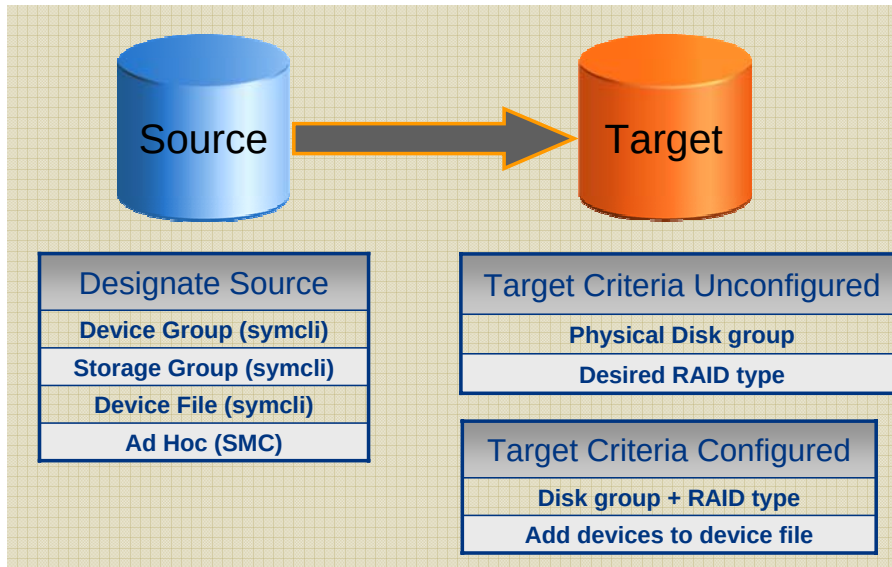


The diagram shows the migration process of an R2 device, which has one remote mirror. When a migration is initiated, the target device occupies the third mirror position during the course of the migration.

The RAID Virtual Architecture (RVA) will allow a maximum of two RAID groups attached as a mirror at the same time. However, this is a transitional state to support operations such as migrations and Optimizer swaps. The array operations associated with the RAID groups will be handled independently for both attached groups. The attached RAID groups may be in different disk groups containing different capacities and speeds.

New Solutions Enabler 7.0 Command

```
symmigrate -name mig_01 -f dev_pairs.txt establish
```



© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 6

Mercer Road
EMC Education Services

Solutions Enabler 7.0 introduces a new SYMCLI command, **symmigrate**, which can be used to perform and monitor migrations. When performing a migration you must designate the source which is the device the data will be migrated from. The device can be selected from the Symmetrix Management Console or on the command line by device group, an auto provisioning storage group, or by a device file that contains a single column listing only the desired source devices.

Next you will need to identify the target which is the volume the data will be migrated to. The criteria and syntax for designating the target will vary based on whether the target is unconfigured or configured. A symcli device group cannot be designated as a target.

When working with configured space you can control which source volumes are migrated to which target volumes by creating a device file with two columns. The first column containing the source device numbers and the second column containing the desired target device numbers.

Migrations are submitted and managed as sessions, so you can use a session name using the **-name** option. Note that the control host for the migration must be locally connected to the Symmetrix V-Max array on which the migration is being performed.

Symmigrate Operations

Control	Action
validate	Tests the user input to see if it would succeed at the current point-in-time
establish	Creates the migration session and will start the synchronization process
terminate	Removes a specific session by session name

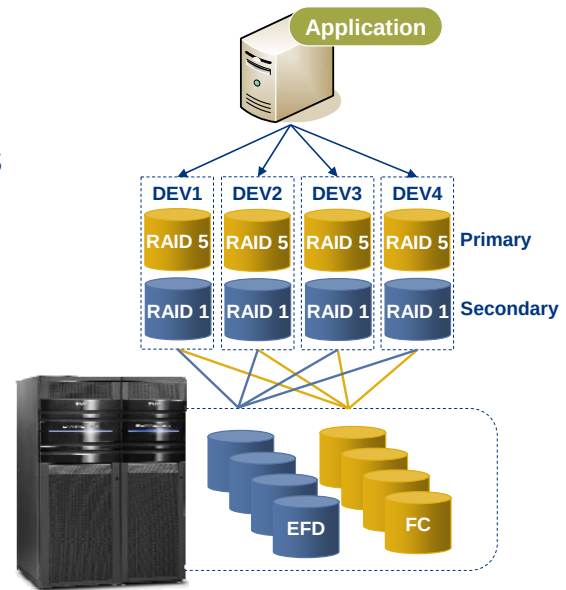
Monitor	Action
query	Provides status about a specific session by session name
list	Shows all sessions for a given Symmetrix array or all local Symmetrix arrays
verify	Determines if a specific session by session name is in a specified state

The symmigrate command has three control actions and three monitor actions. The control actions include validate which tests the user input to see if it will run, establish which creates the migration session and start the synchronization process, and terminate which removes a session by name. The terminate action can only be performed when all volumes in the session are in the Migrated state.

The monitor actions include query which provides status about a specified session, list which shows all sessions for a given Symmetrix array or all local Symmetrix arrays, and verify which determines if the specified session is in a specified state.

Implementation

- Migration facilitated by the creation and movement of RAID groups
- Target RAID group added as secondary mirror
- Target RAID group then made primary mirror
- Upon completion of the migration the original RAID group is either deleted or an iVTOC is performed



© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 8

Mercer Road
EMC Education Services

V-LUN migration is made possible by the creation and movement of RAID groups. Initially the target RAID group is added as a secondary mirror. Once the migration completes, the target RAID group is made the primary mirror. If migrating to unconfigured space, the target RAID group, i.e. the original source device is deleted. If migrating to configured space the target device, i.e. the original source device is iVTOC'd after the migration is completed.

Input Parameters

Source devices

- Device list
- Device group
- Storage group

Configured target devices

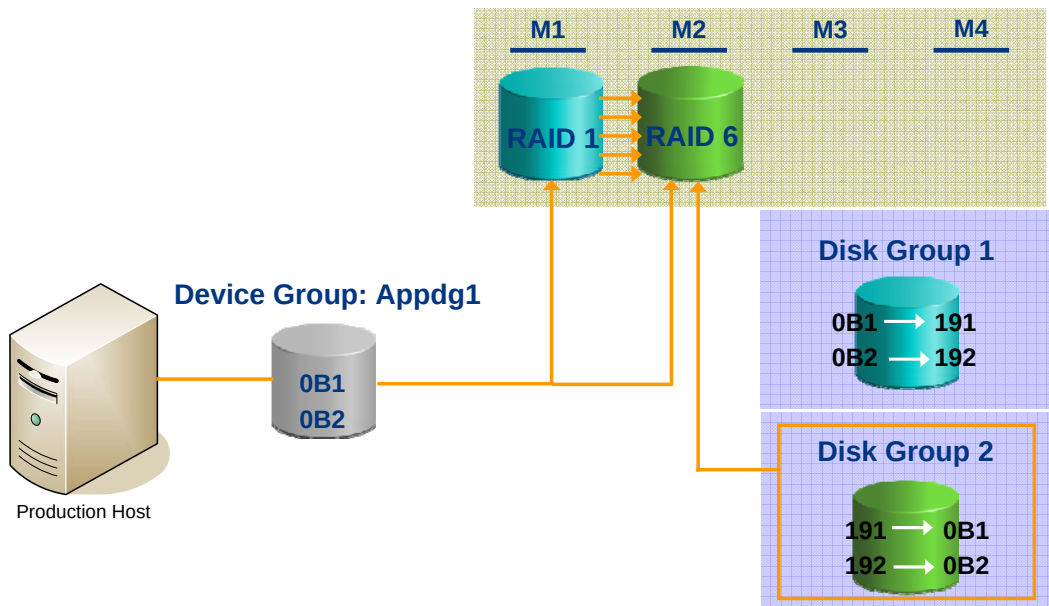
- Device list
 - Exact pairings specified
- Physical disk group
 - Targets selected by Enginuity

Unconfigured target devices

- Physical disk group
 - Target RAID groups built by Enginuity

V-LUN Migration can be controlled either from the command line using the symmigrate command or from the SMC GUI.

Virtual LUN Migration Using Configured Space - 1



© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 10 Mercer Road
EMC Education Services

In the first example we will demonstrate a Virtual LUN Migration which transfers two volumes non-disruptively from RAID-1 to RAID-6 protection. The RAID-1 volumes are part of disk group 1 and the RAID-6 volumes are in disk group 2. When the LUN migration starts the RAID-6 volume temporarily takes up a mirror position while the two volumes are synchronized. Once the migration is complete the original target device assumes the identity of the source and the original source assumes the identity of the target. Data on the target LUN, i.e. the original source is destroyed through IVTOC.

Virtual LUN Migration using Configured Space - 2

```
DMX800SUN3/usr/sengupta> symld -g Appdg list
```

```
Device Group (DG) Name: Appdg
```

```
DG's Type : REGULAR
```

```
DG's Symmetrix ID : 000194900182
```



Standard Device Name			Directors		Device		
Logical	Physical	Sym	SA :P	DA :IT	Config	Att Sts	Cap (MB)
DEV001	c3t50000972C002D*	00B1	08F:0	07B:D0	2-Way Mir	RW	1078
DEV002	c3t50000972C002D*	00B2	08F:0	07D:D0	2-Way Mir	RW	1078

```
DMX800SUN3/> symdisk list -da 7B -interface D -tid 0 -v
```

```
Symmetrix ID : 000194900182
```

```
Disks Selected : 1
```

```
Director : DF-7B
```

```
Interface : D
```

```
Target ID : 0
```

```
Disk Group Number : 1
```

We'll start with a device group called Appdg. It contains two devices 1078 MB in size. Note that the back end of these devices are connected to directors 7B:D0 and 7D:D0. A look at the symdisk output shown on the slide and in the notes section tells us that both of the devices are in disk group 1

```
DMX800SUN3/> symdisk list -da 7D -interface D -tid 0 -v
```

```
Symmetrix ID : 000194900182
```

```
Disks Selected : 1
```

```
Director : DF-7D
```

```
Interface : D
```

```
Target ID : 0
```

```
Disk Group Number : 1
```

In real life environments, a disk group usually comprises disks of the same type of size and performance. Devices with different performance characteristics are placed in different disk groups. Symmetrix devices do not cross disk groups, because that would interfere with tiering storage. Here, we can conclude that both devices reside on disk group 1.

Virtual LUN Migration using Configured Space - 3

```
DMX800SUN3/> symmigrate validate -nop -outfile devlist -g Appdg -tgt_raid6 -
tgt_prot 6+2 -tgt_dsk_grp 2 -tgt_config -name migdemo1
```

```
'Validate' operation execution is in progress for
the device group 'Appdg'. Please wait...
```

```
Validate Migration.....Started.
Validate Migration.....Done.
```

```
'Validate' operation successfully executed for
the device group 'Appdg'.
```

```
DMX800SUN3/> cat devlist
```

```
b2 0192
b1 0191
```

The validate command queries the Symmetrix to find if there are devices that suit the criteria that we asked for, namely, RAID-6 devices with 6+2 protection. The `-tgt_config` option specifies that the migration should be undertaken to configured space. If valid targets exist the device pairing should be written out to a file. This file can later be used as input to the `symmigrate` command.

Virtual LUN Migration using Configured Space - 4

```
DMX800SUN3/> symdev list -range 191:192 -sid 82
```

```
Symmetrix ID: 000194900182
```

Device Name		Directors		Device			
Sym	Physical	SA	:P DA :IT	Config	Attribute	Sts	Cap (MB)
0191	Not Visible	???:?	07A:C4	RAID-6	N/Grp'd	RW	1078
0192	Not Visible	???:?	07D:D4	RAID-6	N/Grp'd	RW	1078

```
DMX800SUN3/> symdisk list -da 7A -interface C -tid 4 -sid 82 -v
```

```
Symmetrix ID : 000194900182
```

```
Disks Selected : 1
```

```
Director : DF-7A
```

```
Interface : C
```

```
Target ID : 4
```

```
Disk Group Number : 2
```

Here we take a look at the two devices suggested by the output of the validate command.

A little known fact is that interrogation marks in the SA column in the `symdev list` output mean the devices are unmapped. Asterisks mean that the devices are mapped to more than one front end port.

There happen to be two RAID-6 devices in this Symmetrix that are not mapped to any director as evidenced by the interrogation marks in the SA column. The devices in the output are connected to backend directors 7A:C4 and 7D:D4.

The devices are in disk group 2. They are suitable targets for our migration.

Virtual LUN Migration using Configured Space - 5

```
DMX800SUN3/> symmigrate establish -sid 82 -file devlist -nop -name migdemo1
'Establish' operation execution is in progress for
the device list in device file 'devlist'. Please wait...
  Establish Migration.....Started.
  Establish Migration.....In Progress.
  Establish Migration.....Done.
'Establish' operation successfully executed for
the device list in device file 'devlist'.
DMX800SUN3/usr/sengupta/Widesky/Cfgmgr> symmigrate list -sid 82 -detail
```

Symmetrix ID: 000194900182

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	TGT Dsk Grp	TGT Protection	Session Name
00B1	0191	12540	SyncInProg	27	02 RAID-6(6+2)		migdemo1
00B2	0192	14964	SyncInProg	13	02 RAID-6(6+2)		migdemo1
Total		-----					
Tracks		27504					
MB(s)		1719.0					
Done(%)		20					

The file devlist is then used as input to a symmigrate command. Once the command is underway a list command shows how far the migration has progressed.

Virtual LUN Migration using Configured Space - 6

```
DMX800SUN3/> symmigrate query -name migdemo1 -sid 82
```

```
Symmetrix ID: 000194900182
```

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
00B1	0191	0	Migrated	100	migdemo1
00B2	0192	0	Migrated	100	migdemo1

```
DMX800SUN3/> symlld -g Appdg list
```

```
Device Group (DG) Name: Appdg
```

```
DG's Type : REGULAR
```

```
DG's Symmetrix ID : 000194900182
```

Standard Device Name			Directors	Device	Cap (MB)
Logical	Physical	Sym	SA :P DA :IT Config	Att Sts	
DEV001	c3t50000972C002D*	00B1	08F:0 07A:C4 RAID-6	RW	1078
DEV002	c3t50000972C002D*	00B2	08F:0 07D:D4 RAID-6	RW	1078

The query output shows slightly different information. We see here that the migration is complete. A listing of the device group shows that devices B1 and B2 are now RAID-6 protected and the back end directors are 7A:4 and 7D:4

Virtual LUN Migration using Configured Space - 7

```
DMX800SUN3/usr/sengupta/Widesky/Cfgmgr> symdev list -range 191:192 -sid 82
```

```
Symmetrix ID: 000194900182
```

Device Name		Directors		Device		
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)
0191	Not Visible	???:? 07B:D0	2-Way Mir	N/Grp'd	RW	1078
0192	Not Visible	???:? 07D:D0	2-Way Mir	N/Grp'd	RW	1078

```
DMX800SUN3/> symmigrate -name migdemo1 -sid 82 terminate -nop
```

```
'Terminate' operation execution is in progress for
the session named 'migdemo1'. Please wait...
```

```
Terminate Migration.....Started.
```

```
Terminate Migration.....Done.
```

```
'Terminate' operation successfully executed for
the session named 'migdemo1'.
```

The old devices 191 and 192 have now become RAID-1 devices connected to backend directors 7B:D0 and 7D:D0. The data on the old physical devices, which are now 191 and 192 is no longer available as the logical volumes have gone through a VTOC.

Finally, the session can be terminated.

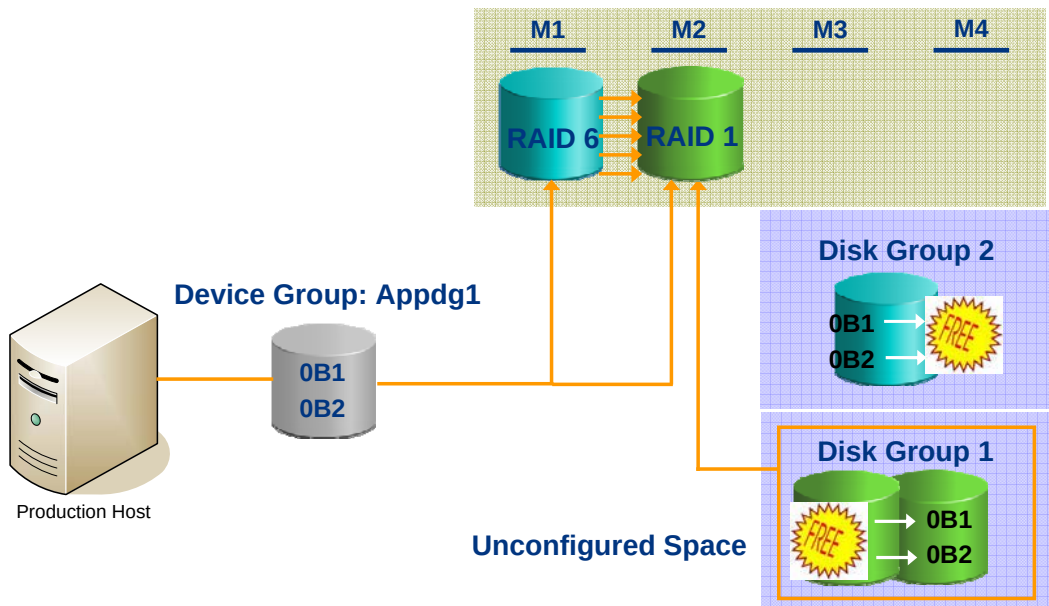
Migration To Configured Space: Summary

Step	Description
1	An available target Symmetrix LUN is determined with a RAID group of the protection and disk type specified.
2	The RAID group is disassociated from the target Symmetrix LUN and is associated to the Symmetrix LUN being migrated as the secondary mirror.
3	The secondary mirror is synchronized with the primary mirror.
4	The primary and secondary indicators are swapped between the two mirrors.
5	The secondary mirror, which is now pointing to the original RAID group, is disassociated from the Symmetrix LUN being migrated and associated as the primary mirror to the target Symmetrix LUN.
6	The target Symmetrix LUN is then iVTOCd to clear the data from it.

Let's review the migration process we observed in the previous demonstration. After specifying configured space and the target protection (or disk type), the establish action instructs the array to perform the following steps:

1. An available target Symmetrix LUN is determined with a RAID group of the protection and disk type specified.
2. The RAID group is disassociated from the target Symmetrix LUN and is associated to the Symmetrix LUN being migrated as the secondary mirror.
3. The secondary mirror is synchronized to the primary mirror.
4. The primary and secondary indicators are swapped between the two mirrors.
5. The secondary mirror, which is now pointing to the original RAID group, is disassociated from the Symmetrix LUN being migrated and associated as the primary mirror to the target Symmetrix LUN.
6. The target Symmetrix LUN is then iVTOC'd to clear the data from it.

Virtual LUN Migration Using Unconfigured Space



© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 18 Mercer Road
EMC Education Services

This time we'll perform a migration from the same device group to unconfigured space.

The unconfigured space to which the data is being moved is designated on the command line with the option `-tgt_unconfigured`. When migrating to unconfigured space we must designate the desired target configuration and protection on the command line, in this case we are designating RAID 1 which will be built automatically and placed in the next available mirror position.

Once we issue the command, the migration session is established and the source is synchronized with the target, while also servicing production I/O from the host. After synchronization completes, the target assumes the identity of the source. The source is then deallocated and the storage capacity is freed up in the original disk group. From the host's point-of-view this operation is completely transparent. Keep in mind that migrating to unconfigured space is a slightly longer operation as the target configuration is created before the migration begins.

Virtual LUN Migration using Unconfigured Space - 1

```
DMX800SUN3/usr/sengupta> symmigrate validate -g Appdg -tgt_raid1 -tgt_dsk_grp 1 -
tgt_unconfig -name migidemo2
```

Execute 'Validate' operation execution for

the device group 'Appdg' (y/[n]) ? y

'Validate' operation execution is in progress for

the device group 'Appdg'. Please wait...

Validate Migration.....Started.

Validate Migration.....Done.

'Validate' operation successfully executed for

the device group 'Appdg'.

```
DMX800SUN3/usr/sengupta> symmigrate establish -g Appdg -tgt_raid1 -tgt_dsk_grp 1
-tgt_unconfig -name migidemo2 -nop
```

'Establish' operation execution is in progress for

the device group 'Appdg'. Please wait...

Establish Migration.....Started.

Establish Migration.....In Progress.

Establish Migration.....Done.

'Establish' operation successfully executed for

the device group 'Appdg'.

This time we will move devices B1 and B2 from the device group Appdg back to disk group 1 with RAID-1 protection. We first validate if this will work and execute the migration. The session takes a little longer to create because the Symmetrix has to search for suitable space.

Virtual LUN Migration using Unconfigured Space - 2

```
DMX800SUN3/> symmigrate -name migdemo2 query -sid 82
```

```
Symmetrix ID: 000194900182
```

Src	Tgt	Invalid Tracks	Status SRC => TGT	Done (%)	Session Name
00B1	N/A	0	Migrated	100	migdemo2
00B2	N/A	0	Migrated	100	migdemo2

```
DMX800SUN3/> symld -g Appdg list
```

```
Device Group (DG) Name: Appdg
```

```
DG's Type : REGULAR
```

```
DG's Symmetrix ID : 000194900182
```

Standard Device Name			Directors		Device			Cap (MB)
Logical	Physical	Sym	SA :P DA :IT	Config	Att	Sts		
DEV001	c3t50000972C002D*	00B1	08F:0 08C:D0	2-Way Mir	RW		1078	
DEV002	c3t50000972C002D*	00B2	08F:0 08A:D0	2-Way Mir	RW		1078	

© 2009 EMC Corporation. All rights reserved.

Virtual LUN Technology - 20



As can be seen devices B1 and B2 are back to being 2-Way-Mir or RAID-1 protected devices

Virtual LUN Migration using Unconfigured Space - 3

```
DMX800SUN3/> symdisk list -da 8C -interface D -tid 0 -sid 82 -v
```

```
Symmetrix ID      : 000194900182  
Disks Selected    : 1
```

```
Director          : DF-8C  
Interface         : D  
Target ID        : 0  
Disk Group Number : 1
```

```
DMX800SUN3/> symdisk list -da 8A -interface D -tid 0 -sid 82 -v
```

```
Symmetrix ID      : 000194900182  
Disks Selected    : 1
```

```
Director          : DF-8A  
Interface         : D  
Target ID        : 0  
Disk Group Number : 1
```

```
DMX800SUN3/> symmigrate terminate -name migdemo2 -sid 82 -nop
```

```
'Terminate' operation successfully executed for  
the session named 'migdemo2'.
```

B1 and B2 are again part of disk group 1.



Migration To Unconfigured Space: Summary

Step	Description
1	A new RAID group of the specified protection type is created on the specified disk type.
2	The newly created RAID group is associated to the Symmetrix LUN being migrated as the secondary mirror.
3	The secondary mirror is synchronized with the primary mirror.
4	The primary and secondary indicators are swapped between the two mirrors.
5	The secondary mirror, which is now pointing to the original RAID group, is deleted.

After specifying unconfigured space and the target protection (or disk type), the establish action instructs the array to perform the following steps:

1. A new RAID group of the specified protection type is created on the specified disk type.
2. The newly created RAID group is associated to the Symmetrix LUN being migrated as the secondary mirror.
3. The secondary mirror is synchronized to the primary mirror.
4. The primary and secondary indicators are swapped.
5. The secondary mirror, which is now pointing to the original RAID group, is deleted.



Supported and Unsupported Features

Supported	NOT Supported
Between protection levels: RAID 1, RAID 5 and RAID 6 protection	Within one disk group, if RAID protection level also does not change
Between drive types: Flash drives, Fibre Channel and SATA disks	Changing device size or meta member size and count
• Protected • Unprotected • Meta devices • FBA, CKD, System i	• VDEVs (TimeFinder/Snap) • SAVE devices • TDEVs (Virtual Provisioning) • TDATs (Virtual Provisioning Thin Pool) • Vault/SFS devices
Active RDF devices (source and target), Active Open Replicator control devices can be used as migration source Clone (source and target), Snap (source) can be migrated	None of the devices listed on left can be a migration target

Interoperability considerations for the Virtual LUN migration feature are shown here. First, Virtual LUN migration requires that either the RAID protection level, the disk group, or both change. You can migrate between device types which is key to supporting intra-array storage tiering, notice that the device size cannot change and when migrating meta devices the size and number of members must remain the same. Also note, CKD striped meta devices can only migrate to other CKD striped meta devices, and that target protection type is restricted to RAID-1. Most device types are supported, note that TimeFinder/Snap and Virtual Provisioning devices are not supported. As should be expected, you cannot migrate from protected to unprotected storage.

Module Summary

Key points covered in this module:

- Virtual LUN migration benefits
- Requirements for Virtual LUN migration
- Virtual LUN migration operational flow
- Host considerations for migrating volumes
- Use of the symmigrate command to perform a migration

These are the key points covered in this module. Please take a moment to review them.

Module 6: Virtual Provisioning Concepts and Planning

Upon completion of this module you will be able to:

- Provide an overview of virtual provisioning
- List the Benefits of Virtual Provisioning
- Describe the components of a Virtual Provisioning environment
- Describe suitable and unsuitable environments for Virtual Provisioning
- Explain the guiding principles behind Thin pool creation and expansion
- List the tools to monitor Thin pool storage space
- Describe local and remote replication capabilities of Thin Devices



These are the objectives for this module. Please take a moment to read them.

Lesson 1: Virtual Provisioning Concepts

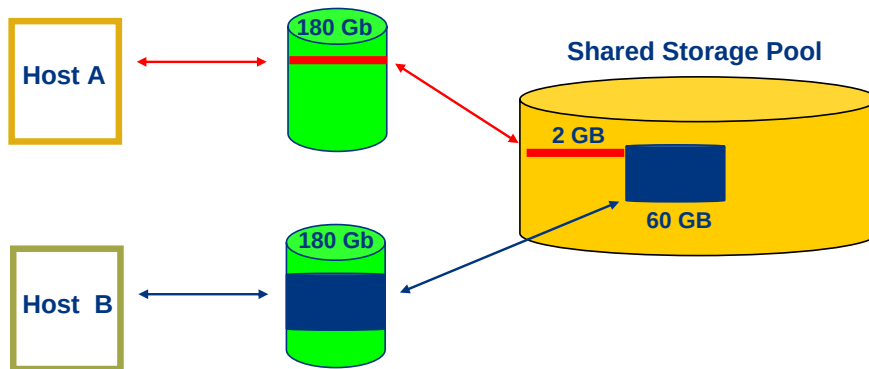
Upon completion of this lesson, you will be able to:

- Provide an overview of virtual provisioning
- List the Benefits of Virtual Provisioning
- Describe the components of a Virtual Provisioning environment



These are the objectives for this lesson. Please take a moment to read them.

Host / Application view of Virtual Provisioning



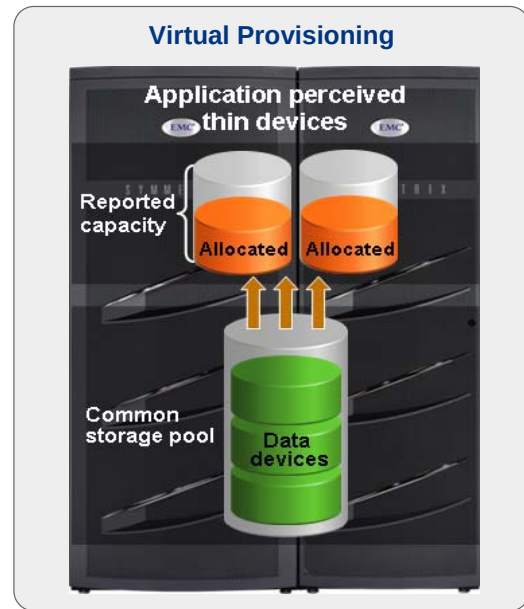
The basic building blocks of Virtual Provisioning are a cache only device called a Thin Device and a Data Device that is part of a shared pool of storage called the thin pool.

In this example Host A has been given access to a 180 Gb logical volume that was created on a thin device. The host treats the device as any other device. Until an application on host A writes to this thin device no physical storage is consumed. If an application on the host writes 2 Mb of data to the volume, 2 Mb of actual storage is used from the devices in the thin pool.

Host B is also mapped to a thin device, which shares storage with Host A. When an application on Host B writes 60 Gb of data to the thin device. 60 Gb of storage will be consumed from the thin pool. Again, storage is only consumed as needed. All the unused storage from the pool is available to be shared.

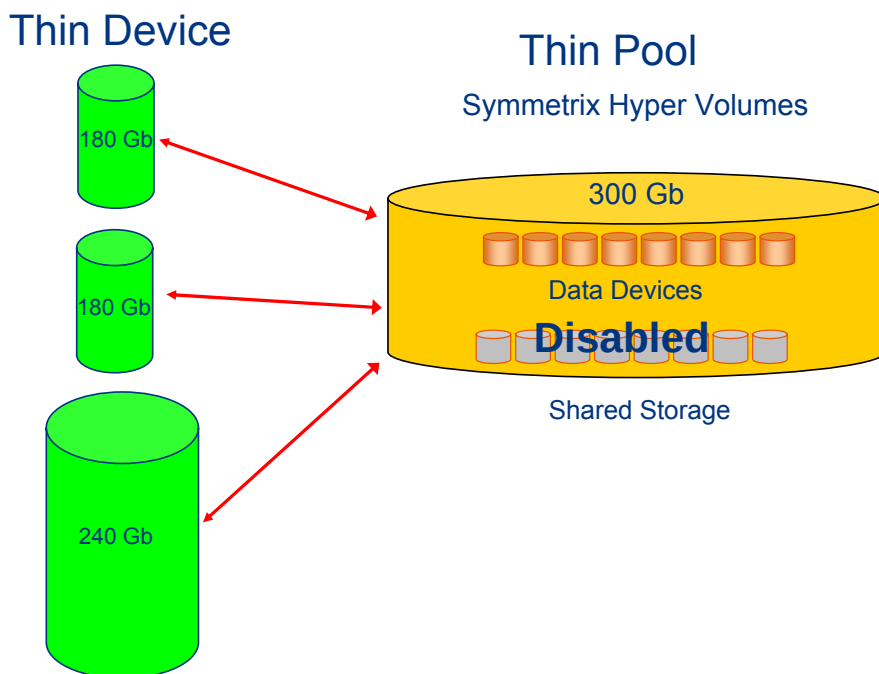
Virtual Provisioning – High Level Description

- Known in the industry as Thin Provisioning
- Large volume presented to a host consumes physical storage from a shared pool only as needed
- Managed and monitored with Symmetrix Management Console (SMC) and the Solutions Enabler command line interface (SYMCLI)



Symmetrix Virtual Provisioning, also known in the industry as “thin provisioning”, is the ability to present a host and therefore an application, with more storage capacity than is physically allocated to it in the storage array. The physical storage is then allocated to the application “on-demand” as it is needed from a shared pool of storage.

Oversubscription of Storage



© 2009 EMC Corporation. All rights reserved.

EMC Confidential – Draft Copy – Do not distribute

Virtual Provisioning Overview - 5

Mercer Road
EMC Education Services

The size of the thin device should be the largest possible size that an application will potentially ever use. This will allow for the future growth of an application and reduce the need to reconfigure the size of the thin device. If the thin device is sized correctly there should be no need to take a host or application off line.

There is no limit on the number of data devices that can be added to a Thin Pool. When a thin device writes to a Thin Pool the data is striped across the data devices. The more data devices in the pool, the wider the striping. This could be a performance benefit for some applications.

Another benefit of virtual provisioning is the ability to over provision storage. In the example there are three thin devices with a total capacity of six hundred gigabytes sharing a three hundred gigabyte Thin Pool. This is called oversubscription. This allows for future growth of an application.

As the oversubscribed Thin pool fills up additional data devices can be added to the pool.

When data devices are added to the Thin Pool they should be added in groups. The reason for this is due to the fact the data is striped across multiple devices. If only one data device is added this could cause a reduction in performance.

Benefits of Virtual Provisioning -1

- Improved capacity utilization
 - Reduce the amount of allocated but unused physical storage
 - Avoid over allocation of physical storage to applications
 - Reduces energy consumption and footprint
- Ease and speed of provisioning
 - Provision independently of physical storage infrastructure
 - Minimize the challenges of growth and expansion
 - Simplifies data layout
 - Saves costs by simplifying procedures to add new storage

One of the issues with the way provisioning is typically done today is the amount of allocated but unused storage. When a host is allocated a lun it has exclusive access to that lun. Any unused space on the lun can not be accessed by any other host. It becomes unused storage. With Symmetrix virtual provisioning, storage is allocated from a shared Thin pool. Multiple host/application can access this shared storage. This means the utilization of storage capacity can be optimized when using Symmetrix Virtual Provisioning. There will be less unused storage.

Another advantage is the ease and speed of provisioning, specifically re-provisioning. While the initial provisioning process is the same for thin devices as it is for regular “thick devices”, adding additional capacity is simply a matter of adding data devices to the Thin pool. By presenting to a host a larger device than is initially required, re-provisioning operations are not required at the host level.

Thus, Symmetrix Virtual Provisioning can simplify storage management. Storage provisioning can be done independently of the physical storage and it can reduce the re-provisioning steps required to support capacity growth.

Benefits of Virtual Provisioning - 2

- Efficiently utilize available resources
 - Wide striping distributes I/O across spindles
 - No requirement for LVM striping or striped metas
 - Reduces disk contention and enhances performance
 - Maximize return on investment

An added benefit is improved performance for certain workloads. There is the overhead of the initial allocation, however because the Thin pool utilizes wide striping, there is a performance benefit for subsequent access.

Symmetrix Virtual Provisioning: Components

- Thin Device
- Data Device
- Thin Pool

These components take part in various operations such as binding, draining, etc.

Symmetrix Virtual Provisioning uses a new type of Symmetrix device called a Thin Device. It is called a thin device because no actual data is saved on the device. The thin device is presented to a host like any Symmetrix hyper volume. An application, on the host, can use this thin device just like any symmetrix logical volume and write data to it.

Data does not get written to the thin device. It is written to a collection of symmetrix hyper volumes, called data devices.

The data devices are grouped together into a storage pool called a Thin Pool. A Thin Device must be bound to a Thin Pool before it can be used. This Thin Pool provides the shared storage. The storage capacity in the Thin Pool can be shared among multiple host and applications.

The action of binding a thin device to a thin pool makes the device ready and associates storage with the thin device.

Virtual Provisioning Device Type - Thin Device

- Consumes no disk space, consists of data structures in cache
- Mapped and masked to a host like any other device
- Presents a Not Ready status until bound to a thin pool
 - Can be mapped and masked prior to binding
- Physical storage is allocated from a thin pool containing data devices
 - Initial allocation of 12 tracks (768 KB) when bound
 - Additional allocation on write or when pre-allocated
- TDEVS can be replicated
 - Local replication using TimeFinder/Snap and TimeFinder/Clone
 - Remote replication using Open Replicator and SRDF
 - Remote replication using Open Replicator

Thin devices are visible to the host. Data devices are not. They use 143 KB of cache plus 8 KB of cache for each GB of reported device size. Thin devices are operating system agnostic and are mapped and masked in the same manner as a regular Symmetrix.

Thin devices can be mapped and masked prior to being bound to a Thin pool containing data devices. When the device is bound to a Thin pool an initial allocation of twelve 64 KB tracks (768 KB) is made unless a larger quota is pre-allocated. More space is allocated as needed by the host application.

TDEVs can be replicated using EMC Symmetrix local and remote replication products

Virtual Provisioning Device Type – Data Device

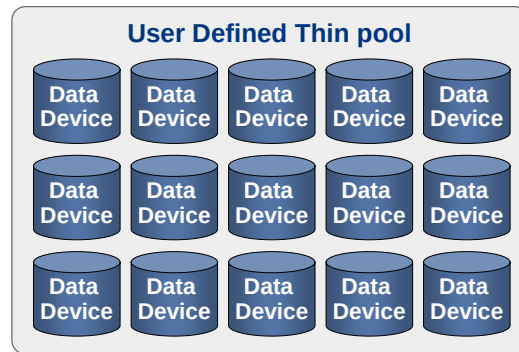
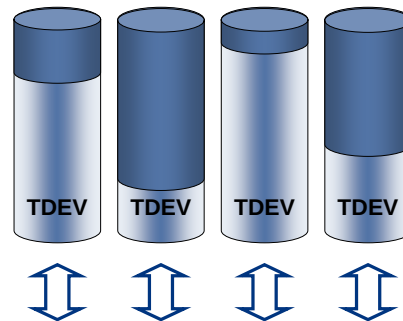
- Non-addressable Symmetrix device
- Provides physical storage for use by a thin device
- Configured to Thin pool to which the thin devices are associated or “bound”
- Data devices should be protected
 - RAID-1 or RAID-6 data devices support local and remote replication
 - RAID-5 clone replication supported in 5773, others in future
- No support for dynamic sparing

Data devices are dedicated to providing storage to thin devices and are not visible to a host. They cannot be mapped, masked, replicated, or configured as metadevices.

Data devices should be protected. If RAID-1 or RAID-6 volumes are used as data devices, local and remote replication is possible. Thin devices bound to a pool containing RAID 5 devices cannot be locally or remotely replicated.

Virtual Provisioning Thin Pools

- Data devices assigned to Thin pools
- Thin pools are user configurable
 - Support tiered storage requirements
 - Work load isolation
- Data devices dynamically added to or subtracted from pool
- Thin pool requirements:
 - Zero or more data devices
 - Need at least one to bind a TDEV
 - Larger pools provide greater efficiency
 - Same protection type (required)
 - Same size (recommendation)



Symmetrix supports three types of save pools; TimeFinder/Snap, SRDF DSE, and Virtual Provisioning Thin pools. Thin pools contain the aggregate available space that is available for a set of thin devices.

A pool can contain zero or more data devices and can be created in a separate operation or at the time that the data devices are created. When a device is added to a pool it can be enabled for allocation or disabled for use in the future.

Unlike Snap Save Pools, there is not a default pool for Thin Devices. Therefore at least one user defined Thin pool must be created. Application performance requirements must be considered as all TDEVs associated with a Thin Pool share the same pool resources, and can compete for access to the same set of spindles. Therefore it is often appropriate to create multiple pools to isolate thin devices by performance and availability requirements. Pools are set up with a specific protection type. This is set when the first data device is added. If first device added to a pool is RAID1, all additional devices must be RAID1.

A maximum of 512 pools of all types combined can exist in the Symmetrix. The most efficient use of pool resources is achieved by using a minimal number of pools so that capacity can be shared.

Binding Thin Devices

- Unbound Thin Devices are not ready for access
- Binding makes thin devices accessible and storage capacity useable
- One device can be bound to one pool only
- Many thin devices can share a pool

Thin devices and data devices can be configured without being bound to a pool at the time of creation.

However, a thin device must be bound to a Thin Pool before it can be used. Binding and unbinding thin devices can be performed using symconfigure commands or SMC. Symmwin can also be used to bind or unbind thin devices to and from a Thin Pool. However, SE and SMC are the recommended methods to bind and unbind thin devices from a pool.

A thin device can only be bound to one Thin Pool at a time. The thin device writes its data to the Thin Pool it is bound to. If a thin device is not bound to a pool, it will be set to user not ready. Once it is bound to a pool it is in the ready state.

Disabling and Unbinding Thin and Data Devices

- Thin Devices can be unbound from pool
 - Must be NR or unmapped before unbinding
 - All data on device is lost
 - Allocation extents on data devices are freed up
- Data devices may be disabled and removed from pool
 - usually after all thin devices using the pool have been unbound

Thin devices can be deleted once they are unbound from the Thin pool. When thin devices are unbound, the extents that have been allocated to them from the Thin pool are freed, causing all data from the thin device to be discarded.

Data devices may also be disabled and removed from a Thin pool. Prior to disabling a data device, all allocated tracks on the data device must be released by unbinding the thin devices that have allocated extents on the data device. Because data on the thin device will be striped across all enabled devices in the pool, in practice all thin devices in a pool will need to be unbound before data devices can be disabled and removed from the pool.

Non-disruptive Removal of Data Devices from Pool

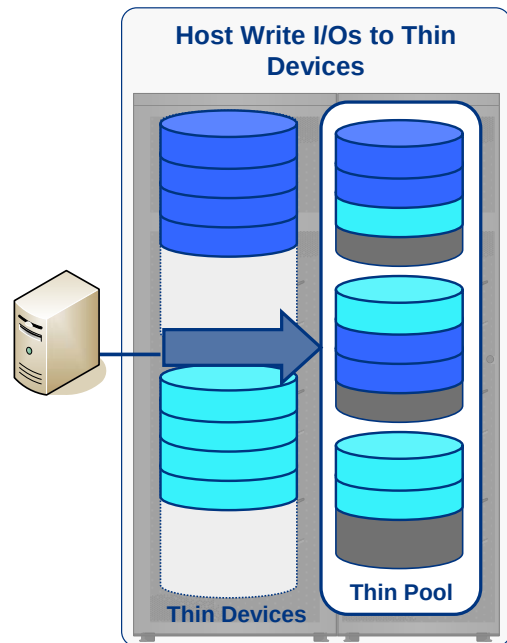
- Device must be disabled
- Disabling causes device to drain
- Used Tracks copied to other data devices

Before removing a data device from a pool, the device must be disabled. Disabling a data device changes its state from Enabled to Draining. The concept of device draining is common to all pool types (data, DSE, and snap pools).

When a device enters a draining state, used tracks on that device are copied to the remaining, enabled devices in the pool. This allows devices to be removed from a pool and repurposed without having to delete any accompanying sessions associated with the device.

Host Writes to Thin Devices

- The initial bind of a thin device to a pool causes 1 thin device extent to be allocated per thin device
- When a write is performed, if there is no free space in a previously allocated extent, a new thin device extent is allocated from the pool
- A round-robin mechanism is used to balance the allocation of DATA device extents across all available DATA devices in the pool
- A 4 member thin meta would cause 4 extents to be allocated

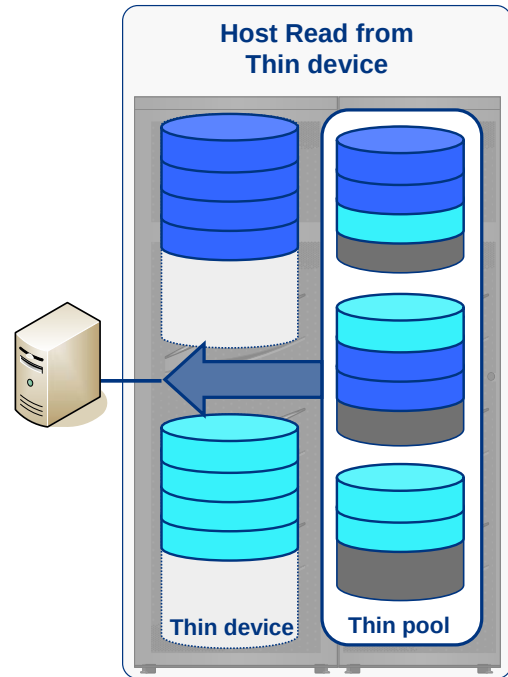


As a host performs write operations, Enginuity will sequentially allocate storage to the thin device from the data devices in the associated pool. The unit of allocation is an extent which is 12 tracks or 768Kbytes. The extents are striped across all members of the pool. This wide striping can potentially provide increased performance.

The allocation is sequential regardless of whether the host writes are sequential or random.

Host Reads From Thin Device

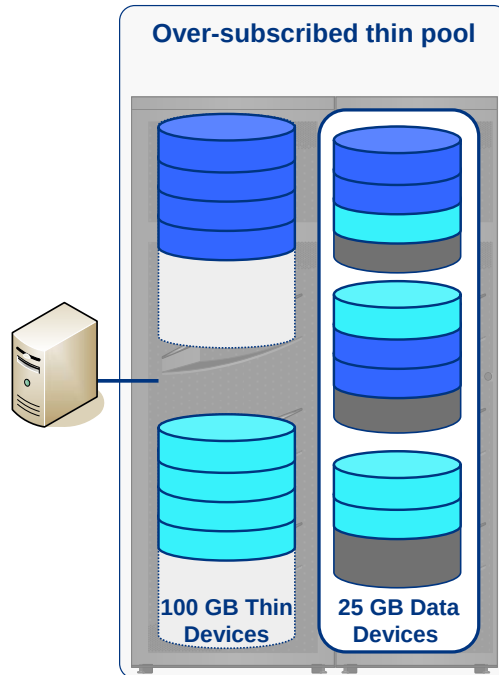
- Thin devices are cache devices and contain only pointers to allocated blocks on DATA devices
- A read from a thin device, retrieves data from the data device
- Read to unallocated blocks
 - returns zeros
 - does not trigger storage allocation



A thin device is seen by the host like any other device. Normally an application will only read storage that was previously written to. However, if the host reads a block that was not previously written to, the Symmetrix returns data blocks that contain all zeros.

Over-subscription

- Bound TDEV capacity is greater than available capacity in the thin pool
 - Considered normal
 - Addresses inefficiency of over allocation but under utilized storage
- Over-subscription does create risk
 - If the pool becomes full, writes to the TDEVs fails and an IO error is returned
 - Storage administrator must monitor pool to prevent “pool full” condition
- Data devices can be dynamically added to the thin pool
 - Best practice is to add multiple devices to the pool to prevent bottlenecks



© 2009 EMC Corporation. All rights reserved.

EMC Confidential – Draft Copy – Do not distribute

Virtual Provisioning Overview - 17

Mercer Road
EMC Education Services

Over-subscription is where the total capacity of all TDEVs bound to a Thin pool is greater than the aggregate capacity of all data devices in the pool. This is a normal practice but does require that a storage administrator monitor the pool to prevent a “pool full” condition. When the pool is full and a host attempts to perform a write that requires a new extent to be allocated, a write error is returned to the host.

Different hosts react differently when this error is experienced. The Host Connectivity Guide for each operating system describes the behavior of the specific operating system, volume manager and file system. These Guides are available on PowerLink

When additional capacity is needed in the pool, data devices can be dynamically assigned to the pool. It is a best practice to add multiple devices to the pool at the same time to ensure that the allocations to a thin device are spread across multiple data devices

To reduce the risk of over subscription, a limit can be set for a pool that would limit the total capacity of the thin devices that are bound to a thin pool as a percent of the pool capacity.

Lesson 2: Planning for Virtual Provisioning

Upon completion of this lesson, you will be able to:

- Describe suitable and unsuitable environments for Virtual Provisioning
- Explain the guiding principles behind Thin pool creation and expansion
- List the tools to monitor pool storage space
- Describe local and remote replication capabilities of Thin Devices



These are the objectives for this lesson. Please take a moment to read them.

VP Friendly Environments

- Applications with tolerance for performance variability are suitable for VP such as:
 - Document and Media Repositories
 - Test Development and QA
- Databases that use “auto-extend” capabilities during page formatting or table creation
- Applications with predictable and controlled growth
- Flash drives
 - Compared to rotating drives, flash drive performance does not degrade as disks fill up

Virtual Provisioning is appropriate for applications for which some performance variability can be tolerated. Some workloads will see performance improvements from wide striping with Virtual Provisioning. However, when multiple thin devices contend for shared spindle resources in a given pool, and when utilization reaches higher levels, the performance for a given application and thin device might become more variable.

Document repositories can be a strong fit for Virtual Provisioning, provided the environments meet the performance, growth, and other criteria outlined previously, as rapidly rising capacity requirements for these repositories create the opportunity for improved capacity.

Many organizations will see an opportunity to lower TCO by improving ease of use and capacity utilization for storage associated with test, development, and QA activities, which often can tolerate some level of performance variability.

Applications with a predictable and controllable growth rate are good fits for VP because they do not run the risk of unexpectedly consuming most or all of the shared storage.

Virtual Provisioning can help ensure that Flash drive resources are highly utilized. Flash drives can offer excellent response times as well as significant IOs/GB. With a traditional drive, larger seek times are encountered during reads as the drive fills up. Read response times in Flash drives do not decline as utilization of the drive increases.

VP Hostile Environments

- Any application that touches storage capacity without using it causes unused space to become fully allocated, such as:
 - File systems (like NTFS) that append data to file system instead of reusing free space
 - Some utilities that copy “raw” data such as “dd” in Unix
 - Administrative operations such as bad block detection or file system checks that cause dense write patterns on all reported storage

There are a variety of factors that can contribute to making a given application environment thin-hostile. One step or a combination of steps involved in simply preparing storage for use by the application may force all of the storage that is being presented to become fully allocated.

If the storage space management policies of the application and underlying software components do not tend to reuse storage that was previously used and released, that will increase the speed in which underlying thin devices become fully allocated.

Data copy operations such as disk balancing operations and de-fragmentation operations that are carried out as part of the administration of the environment.

If there are administrative operations such as bad block detection operations or file system check commands that perform dense patterns of writes on all reported storage.

Configuring Thin Pools

- Different pools for different classes of storage
- Different pools for different application families
- All data devices should reside on drives with the same rotational speed
- All data devices should be spread evenly across as many DAs and drives as possible
- Data devices should be the same size, if possible
 - Using different sizes could result in uneven data distribution

When planning a configuration using thin devices the first step involves determining how many separate Thin pools are needed and the required composition of each thin data pool. Typically this will involve conceptually organizing disk storage into separate classes, with further subdivision as needed to allow the back-end resources (drives and DAs) used by the pools to be isolated from one another. Depending on the mix of applications to be placed on thin devices, it will often be necessary to create multiple thin pools but generally, the most efficient use of resources will be achieved by using a minimal number of pools.

Typically, a Thin pool should be designed for use by a given application, or set of related applications, aligned with a given business group. The applications sharing a Thin pool will compete for back-end resources, including Thin pool storage capacity, so applications should not share the same pool if this is not acceptable. The devices comprising a Thin pool will have the same performance and protection properties, so the applications sharing a Thin pool should have the same performance and protection requirements.

Pool Expansion Requirements

- Managing pool expansion is critical to performance
 - Start with a large pool
 - Expand by same number of disks as original pool
- Must have same protection as devices in target pool
- Must have same emulation as devices in target pool

When an oversubscribed Thin pool begins to run out of space, data devices should be added to the pool before the pool completely fills up. Otherwise there is a risk that an application that performs a write to a previously unmapped portion of a thin device may encounter an out-of-space error. Data devices can be added to a Thin pool non-disruptively. The set of data devices to be added to an existing Thin pool should satisfy the following requirements:

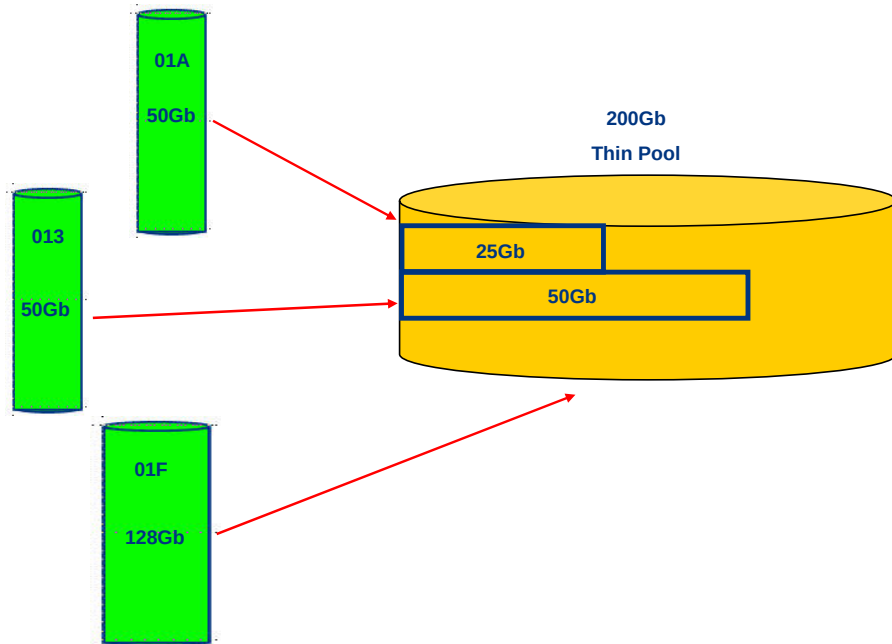
- They must have the same protection type as the devices already in the target Thin pool.
- They must have the same emulation as the devices already in the target Thin pool.

Pool Expansion Recommendations

- Should reside on drives with same RPM
- Should use equal amount of space from underlying drives
- Should use equal size data drives
- Spread devices across DA and physical drives
- To expand effectively:
 - Enable new TDATs on all existing pool drives
 - Will increase capacity with same performance
 - Add a number of drives equal to original pool size
 - Will increase both capacity *and* performance
- ***DO NOT*** add just a few TDATs to a large pool

Typically, the DA and drive configuration underlying a set of data devices to be added to a pool should be sized to have enough available performance capacity to handle the workload that will be delivered to any regions of thin devices that get allocated after the pool addition. This is because once the pool addition is made and any remaining capacity in the pre-existing pool becomes completely exhausted, a point will be reached where all storage allocations made from the pool will be satisfied using just data devices from the pool addition. In most cases it is inadvisable to add a set of data devices that is spread over a small number of drives to a thin pool that is running out of space.

Pre-allocation of Pool Space



A pre-allocated amount of Thin Pool space can be allocated and assigned to the Thin Device. Only the Thin Device assigned to the region can write to it. In this example thin device 01A has 25 gigabytes of allocated and assigned space. Thin device 013 has its entire 50 gigabytes allocated and assigned. Thin device 01F does not have an allocated or assigned space. An application that writes to device 01F could possibly consume up to 128 GB of shared storage. However it could not over write the region the 25 GB or the 50 GB of allocated and assigned space of the other two devices.

Reasons for Space Pre-allocation

- Can sometimes be used in thin hostile environments for performance benefits
- Overcomes latency caused by initial write to an unallocated device

Applications that pre-allocate or pre-write all capacity of their thin devices may still see performance benefits from the pooled storage which tend to maintain a better balance of activity across the configured storage resources.

Pre-allocation also overcomes a slight, temporary overhead on first write to a new extent.

Performance Considerations

- Extent allocation is striped across all enabled data devices in pool
 - With regular Symmetrix volumes, wide striping can be achieved using striped meta devices
 - Meta member devices striped across physical disks
 - Thin devices are widely striped automatically
 - Makes it easier to avoid hotspots
 - Potential performance improvement for some workloads
 - Potentially different applications could share data devices
- Pool-based view of storage layout
 - Thin pools should contain storage of a particular class with a defined business purpose
 - Planning consists largely of deciding what class of storage is required and which business group's resources should be used
 - Storage is viewed at a higher level when planning new devices
 - Think pool, not hyper!

The answer to the question of thin device performance is the classic “it depends”. Benchmarks have shown that thin device performance could be better than regular devices. However, real world application performance is highly dependent on the workload and the workload or other applications that are sharing the Thin pool. Therefore the conventional wisdom is that virtual provisioning is best suited for applications that have “general purpose” performance requirements.

In addition to potential contention for access to shared data device, response time and throughput can be impacted by the overheads incurred the first time a write is performed on an unallocated region of a thin device. This will disappear once the “working set” of a thin device has been written.

Care must be taken when migrating an application whose backend layout has already been tuned to isolate workloads or where an application has stringent response time and/or throughput requirements.

Meta Devices Considerations

- Meta devices are normally created for the following reasons:
 - Increase the size of a volume that is presented to a host
 - Maximum Symmetrix device size limit of 65,472 cylinders (59.98 GB)
 - Increase performance (striped metavolumes)
 - Spread workload over more spindles
- With TDEVs there is no need create Meta devices to spread workload
 - Thin devices already are wide striped across members of the Thin pool
- The Symmetrix limit on device size still applies to TDEVs, therefore meta devices are appropriate when the host needs to see larger devices
 - Best Practice is to configure concatenated
 - Advantages:
 - Concatenated thin meta devices can be expanded while preserving data
 - Non-meta thin devices can be easily converted to concatenated thin meta devices

The same metadvice commands (form meta, dissolve meta, convert meta) can be used to configure a thin meta device.

When creating a thin metadvice, all members must be thin devices. Mixing and matching thin and thick devices is not supported.

Monitoring Thin Pools

- Recommended capacity utilization per pool between 60 and 80%
 - $\approx 2x$ on “thick” devices
- Pools should be configured to absorb data from occasional “runaway” applications
- Monitoring can be done with
 - Symmetrix Management Console (SMC)
 - Solutions Enabler **symcfg monitor**
 - Best to monitor with event daemon **storeventd**

Virtual provisioning simplifies, but does not entirely eliminate, the need for proper planning. When implementing Virtual Provisioning it is important that realistic utilization objectives are set. Generally, organizations should target no higher than 60 percent to 80 percent capacity utilization per pool. A buffer should be provided for unexpected growth or a “runaway” application that consumes more physical capacity than was originally planned for. There should be sufficient free space in the Thin pool equal to the capacity of the largest unallocated thin device.

Thin pool storage can be managed with SMC, Solutions Enabler and the event daemon. The event daemon offers the most flexible way of monitoring free space in a Thin pool.

Pool Full Condition



- Avoid Pull full at all costs by monitoring and taking timely action
- When pool fills up
 - Next write to thin device will fail
 - Data on thin device can still be read
- Different Operating Systems react differently when pool fills up
 - Details in Host Connectivity Guides on PowerLink

Given the nature of Virtual Provisioning, there is always the danger that storage space in a Thin pool is exhausted due to oversubscription. This condition should be avoided at all costs by proper monitoring and taking preventive action. However, should this circumstance occur, this is what happens:

Once space in the pool is exhausted, no new writes to the thin devices are allowed. The reaction to the failed write will depend on the application and the Operating System. The Host Connectivity Guides for all operating systems are available on EMC PowerLink and describe the behavior when the “No space on Device” condition is encountered.

The data in the Thin pool is still available for reads.

Thin Device Local Replication

- Only Thin to Thin replication is supported
- Support available for all RAID types (1, 5 and 6)
- TimeFinder/Snap
 - Performed with Thin Devices sources and VDEV targets
- TimeFinder/Clone
 - Performed with Thin Device STDs and Thin Device targets
- No support for Thin BCVs hence TF/Mirror not supported

Local replication is permitted with thin devices for TimeFinder/Snap and TimeFinder/Clone. Support is available for thin devices that are bound to RAID-1, RAID-5 or RAID-6 protected Thin pools.

The steps to create Thin Device Snaps is like creating snap copies of regular devices. For Clone replication, it would be necessary to create a device group with thin source and thin target devices. Other than that there is no distinction between the local replication procedures with thin and thick devices.

Since Thin devices can only participate in Thin to Thin replication operations and since Thin BCVs are not supported, there is no support for Thin devices in TimeFinder/Mirror configurations.

Thin Device Remote Replication

- Support available for all RAID types (1, 5 and 6)
- SRDF (Sync, Async, Adaptive Copy) – thin to thin only
 - Create Thin Devices
 - Assign the Dynamic RDF attribute to the devices
 - Create Dynamic RDF pairs between thin devices
- Open Replicator
 - Thin devices can act as control or remote device
 - When acting as source, zeroes sent from unallocated space
 - When acting as target, storage space is fully allocated in accordance with size of source

Remote replication is permitted with thin devices for all flavors of SRDF and Open Replicator. Support is available for thin devices that are bound to RAID-1, RAID-5 or RAID-6 protected Thin pools.

Module Summary

Key points covered in this module:

- Overview of virtual provisioning
- Benefits of Virtual Provisioning
- Components of a Virtual Provisioning environment
- Suitable and unsuitable environments for Virtual Provisioning
- Guiding principles behind Thin pool creation and expansion
- Tools to monitor Thin pool storage space
- Local and remote replication capabilities of Thin Devices

These are the key points covered in this module. Please take a moment to review them.

Module 7: Virtual Provisioning with Solutions Enabler

Upon completion of this module you will be able to:

- List the steps to create Thin and Data Devices
- Explain how to add Data Devices in Thin Pools
- Define the process to bind Thin Devices to Thin Pools
- State how to create and bind Thin Meta Devices to a Thin Pool
- Relate how to unbind Thin Devices and free up pool space
- Indicate the process of setting a subscription limit



These are the objectives for this module. Please take a moment to read them.

Lesson 1: VP Implementation using Config Manager

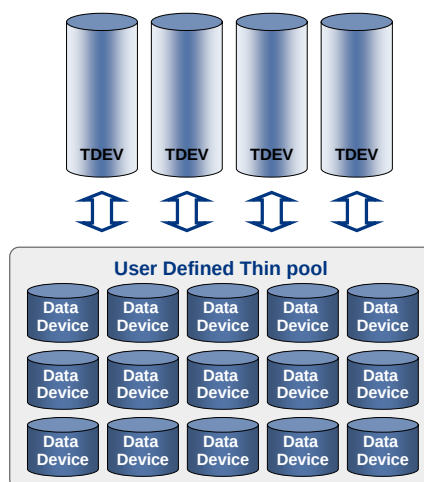
- Upon completion of this lesson, you will be able to
- List the steps to create Thin and Data Devices
- Explain how to add Data Devices in Thin Pools
- Define the process to bind Thin Devices to Thin Pools



These are the objectives for this lesson. Please take a moment to read them.

Implementation Steps

0. Planning
1. Create Thin Devices (TDEV)
2. Create Data Devices
3. Create Thin Pool
4. Add Data Devices to Thin Pool
5. Bind TDEV to pool
6. Map and mask TDEV
7. Monitor TDEV allocation and thin pool



Like all aspects of storage configuration, the first step is planning. However with thin devices less planning is required in that you don't have to think about back-end layout and there is minimal efficiency impact of over estimating the required capacity.

In the next few pages, we will review configuring thin devices, data device, and thin pool Using Solutions Enabler. Once the thin devices are configured, they can be mapped, masked and used like any other device.

Symmetrix Environment for Screen Output

```
sun196/usr/sengupta> 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
000190301138	Local	DMX4-6	5773	8192	3	259
000190301194	Local	DMX4-6	5773	16384	3	259

```
sun196/usr/sengupta> symdev list -all -sid 38
```

Symmetrix ID: 000190301138

Device Name		Directors		Device		
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)
0000	Not Visible	???:? 01A:C0	VAULT	N/A	N/A	5120
0001	Not Visible	???:? 16A:C1	VAULT	N/A	N/A	5120
0061	Not Visible	???:? NA:NA	VDEV	N/Asst'd	NR	1078
0062	Not Visible	???:? NA:NA	VDEV	N/Asst'd	NR	1078
0102	Not Visible	01D:0 01B:C5	2-Way Mir	N/Grp'd	NR	17250

The output on this page is intended to show the Symmetrix environment where the VP operations are going to be performed. Initially Symmetrix 38 will be used for the examples.

Create Thin Devices

```
sun196/usr/sengupta/Vp> symconfigure -sid 38 -file mktdev.cfg prepare -nop -v
```

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

```
Establishing a configuration change session.....Established.
Processing symmetrix 000190301138
{
  create dev count=16, size=1093, emulation=FBA,
    config=TDEV, mvs_ssid=0000;
}
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Submitting configuration changes.....Submitted
Validating configuration changes.....Validated.
```

```

New symdevs: 0103:0112
Initiating PREPARE of configuration changes.....Prepared.
Closing configuration change request.....Closed.
Terminating the configuration change session.....Done.
The configuration change session has completed successfully.
```

```
sun196/usr/sengupta/Vp> symconfigure -sid 38 -file mktdev.cfg -nop commit
```

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

The syntax related to creating thin devices is available in Chapter 1 of the Array Controls Manual.

It is usually a good idea to use the **prepare** option in Config Manager to make sure that the syntax for the command file is in order. Once the **prepare** is successful, the **commit** action makes the necessary changes in the Symmetrix.

The file **mktdev.cfg** contained the following line:

```
create dev count=16, size=1GB, emulation=fba,config=TDEV;
```

The verbose option in the command line shown above translates this input into:

```
{
  create dev count=16, size=1093, emulation=FBA,
    config=TDEV, mvs_ssid=0000;
}
```

In this case 16 Thin Devices were created with addresses ranging from 103 to 112, which are the device numbers at the end of the last device shown on the previous page.

Display Newly Created Thin Devices

- Thin devices will be in a **Not Ready** state until bound to a thin pool
- No back-end resources are allocated

```
sun196/usr/sengupta/Vp> symdev list -tdev -unbound -sid 38
```

```
Symmetrix ID: 000190301138
```

Device Name		Directors			Device		
Sym	Physical	SA :P	DA :IT	Config	Attribute	Sts	Cap (MB)
0103	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0104	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0105	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0110	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0111	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0112	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025

After configuring thin devices they can be viewed using the `symdev list` command. Note that TDEVs are not ready to a host until bound to a thin pool. Also note that DA:IT show NA in that no backend resources are allocated to thin devices.

Create Data Devices

```
sun196/usr/sengupta/Vp> cat mkddev.cfg
create dev count=8, attribute=datadev, size=1GB, emulation=fba, config=2-way-Mir;
```

```
sun196/usr/sengupta/Vp> symconfigure -sid 38 -file mkddev.cfg -nop commit
```

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

```
Establishing a configuration change session.....Established.
Processing symmetrix 000190301138
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Submitting configuration changes.....Submitted
Validating configuration changes.....Validated.
```

```
New symdevs: 0113:011A
Initiating PREPARE of configuration changes.....Prepared.
Initiating COMMIT of configuration changes.....Queued.
```

```
.....
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

It is a performance engineering recommendation that data devices be distributed on separate DA slices. In practice this works if you assign a pristine disk group to the data devices. Symmwin algorithms take many factors into consideration such as how many splits are on a particular physical disk, dual initiator considerations, etc. in laying out data devices. Simply specifying 8 TDATs because there are 8 DA slices is not enough.

Display Data Devices

```
sun196/usr/sengupta/Vp> symdev list -datadev
```

```
Symmetrix ID: 000190301138
```

Device Name		Directors		Device				
Sym	Physical	SA :P	DA :IT	Config	Attribute	Sts	Cap (MB)	
0113	Not Visible	???:?	01A:C4	2-Way Mir	N/A	(DT) RW	1025	
0114	Not Visible	???:?	02B:C6	2-Way Mir	N/A	(DT) RW	1025	
0115	Not Visible	???:?	02A:D6	2-Way Mir	N/A	(DT) RW	1025	
0116	Not Visible	???:?	01B:D4	2-Way Mir	N/A	(DT) RW	1025	
0117	Not Visible	???:?	01A:D3	2-Way Mir	N/A	(DT) RW	1025	
0118	Not Visible	???:?	02B:D1	2-Way Mir	N/A	(DT) RW	1025	
0119	Not Visible	???:?	16A:C1	2-Way Mir	N/A	(DT) RW	1025	
011A	Not Visible	???:?	01B:C1	2-Way Mir	N/A	(DT) RW	1025	

After configuring thin devices they can be viewed using the `symdev list` command. Data Devices are private devices and are not directly visible to a host. Data devices can be identified by the device type of DT.

Create Thin Pool

- Thin pools can be created at the same time as data devices or performed in separate operations
- There is no default thin pool, only user defined pools

```
sun196/ symconfigure -sid 38 -cmd "create pool WB type=thin;" commit -nop
```

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

```
Establishing a configuration change session.....Established.
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Committing configuration changes.....Reordering.
Creating pools .....Done.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

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

Thin pools can be created at the same time as creating the DATA devices.

Unlike SAVE pools used with TimeFinder/Snap or SRDF/DSE, there is no default thin pool. Therefore at least one user defined thin pool must be created before Virtual Provisioning can be used.

The –cmd option specifies the list of commands containing the configuration change definitions on the command line. Each command is terminated by a semicolon.

Add Data Devices to Pool

- Data devices can be added to a thin pool at time of creation
- A data device must be enabled before being used for allocation
 - Can be performed in a separate operation

```
sun196/usr/sengupta/Vp> symconfigure -sid 38 -cmd "add dev 113:114 to pool WB
type=thin, member_state=ENABLE;" commit -nop
```

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

```
Establishing a configuration change session.....Established.
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.

Add data devices to the pool and enable them so that they can be used

Display Thin Pool

- State of thin pool is Enabled as soon as one data device is enabled
- Currently no devices are bound and thus no tracks have been allocated

```
sun196/usr/sengupta/Vp> symcfg list -pool -thin -sid 38 -mb
```

```
Symmetrix ID: 000190301138
```

S Y M M E T R I X P O O L S									
Pool Name	Type	Device Emul	Device Config	Total MBs	Enabled MBs	Used MBs	Free MBs	Full (%)	Status
WB	TH	FBA	2-Way Mir	2049	2049	0	2049	0	Ena
Total MBs				2049	2049	0	2049	0	

Legend for Pool Types:

SN = Snap, RS = Rdfa DSE TH = Thin

This pool is identified as an enabled Thin Pool. The pool is enabled as soon as it contains one enabled data device. The output above can also be displayed in tracks.

Detailed Information about Pool

```
sun196/usr/sengupta/Vp> symcfg show -pool WB -thin -sid 38 -mb
```

```
Symmetrix ID: 000190301138
```

```
Symmetrix ID           : 000190301138
Pool Name              : WB
Pool Type              : Thin
Dev Emulation          : FBA
Dev Configuration      : 2-Way Mir
Pool State             : Enabled
# of Devices in Pool   : 2
# of Enabled Devices in Pool : 2
```

```
Enabled Devices(2):
```

```
{
```

Sym Dev	Total MBs	Used MBs	Free MBs	Full (%)	Device State	Session Status
0113	1025	0	1025	0	Enabled	N/A
0114	1025	0	1025	0	Enabled	N/A
MBs	2049	0	2049	0		

```
}
```

The detailed information on the pool displays which devices are participating in the pool. There are two options that can be used with the **SHOW** command to display more information about the pool. The **-all** option shows information on enabled and disabled pools, while the **-detail** option includes a listing of the thin devices.

Bind Thin Devices to Thin Pool

- TDEV must be bound to a thin pool before use
 - May be bound to a thin pool at time of creation
- The first extents are allocated to the TDEV when bound
 - Preallocation reduces risk of failure due to over subscription

```
sun196/ symconfigure -sid 38 -cmd "bind tdev 103:106 to pool WB;" commit -nop
A Configuration Change operation is in progress. Please wait...
```

```
Establishing a configuration change session.....Established.
Processing symmetrix 000190301138
Performing Access checks.....Allowed.
Checking Device Reservations.....Allowed.
Locking devices.....Locked.
Validating configuration changes.....Validated.
Preparing configuration changes.....Prepared.
Committing configuration changes.....Started.
Binding devices.....Done.
Committing configuration changes.....Committed.
Terminating the configuration change session.....Done.
```

The configuration change session has successfully completed.

If thin devices were to be bound to a pool at the time of creation, the syntax would read as follows:

```
create dev count=2, size=1GB,
emulation=FBA, config=TDEV,
binding to pool WB, preallocate size=1 GB;
```

Bound Devices

- After binding, one extent (12 Tracks) are allocated to each TDEV

```
sun196/usr/sengupta/Vp> symcfg list -tdev -sid 38
```

```
Symmetrix ID: 000190301138
```

```
Enabled Capacity (Tracks) :      32784
```

```
Bound Capacity (Tracks) :      65580
```

SYMMETRIX THIN DEVICES

Sym	Pool Name	Dev Emul	Total Tracks	Pool Subs (%)	Pool Allocated Tracks (%)	Pool Written Tracks (%)	Status
0103	WB	FBA	16395	50	12 0	- -	Bound
0104	WB	FBA	16395	50	12 0	- -	Bound
0105	WB	FBA	16395	50	12 0	- -	Bound
0106	WB	FBA	16395	50	12 0	- -	Bound
0107	-	FBA	16395	0	0 0	- -	Unbound
0112	-	FBA	16395	0	0 0	- -	Unbound
Total Tracks			262320	800	48 0	- -	

In the prior page we bound 4 TDEVs to the pool WB. As a result, one extent, or 12 tracks were allocated to each TDEV. Also the TDEVs are now ready and available for Read Write operations once they are mapped and masked to a host.

Thin Pool After Binding TDEVs – Data Devices

```
sun196/symcfg show -pool WB -thin -detail -sid 38
```

```
Symmetrix ID: 000190301138
```

```
Symmetrix ID      : 000190301138
Pool Name         : WB
Pool Type         : Thin
Dev Emulation     : FBA
Dev Configuration : 2-Way Mir
Pool State        : Enabled
# of Devices in Pool : 2
# of Enabled Devices in Pool : 2
Max. Subscription Percent : None
```

```
Enabled Devices(2):
```

```
{
-----
Sym   Total  Alloc  Free Full Device
Dev   Tracks Tracks Tracks (%) State
-----
0113   16392    24  16368  0 Enabled
0114   16392    24  16368  0 Enabled
-----
Tracks  32784    48  32736  0
}
```

```
(....continued on right column)
```

```
(....continued from left column)
```

```
Thin Devices(4):
```

```
{
-----
Sym   Pool   Pool   Pool
Dev   Total Subs Allocated Written
      Tracks (%) Tracks (%) Tracks (%) Status
-----
0103   16395   50    12  0 - - Bound
0104   16395   50    12  0 - - Bound
0105   16395   50    12  0 - - Bound
0106   16395   50    12  0 - - Bound
-----
Tracks  65580  200    48  0 - -
```

The detailed output shows the correct percentage subscription of the thin devices compared to the pool. Displaying the storage space in tracks gives a more accurate representation than in MB. The same output in MB would show:

```
sun196/usr/sengupta> symcfg show -pool WB -thin -detail -sid 38 -mb
```

```
Enabled Devices(2):
```

```
{
-----
Sym   Total  Alloc  Free Full Device
Dev   MBs    MBs    MBs (%) State
-----
0113   1025    2    1023  0 Enabled
0114   1025    2    1023  0 Enabled
-----
MBs    2049    3    2046  0
}
```

```
Thin Devices(4):
```

```
{
-----
Sym   Pool   Pool   Pool
Dev   Total Subs Allocated Written
      MBs (%) MBs (%) MBs (%) Status
-----
0103   1025   50    1  0 - - Bound
0104   1025   50    1  0 - - Bound
0105   1025   50    1  0 - - Bound
0106   1025   50    1  0 - - Bound
-----
MBs    4099  200    3  0 - -
```

TDEVs after Mapping and Masking

```
sun196/usr/sengupta> sympd list -sid 38
```

```
Symmetrix ID: 000190301138
```

Device Name	Directors			Device				Cap
Physical	Sym	SA	:P DA :IT	Config	Attribute	Sts	(MB)	
/dev/rdisk/c3t16d0s2	0060	02C:0	01A:D5	2-Way Mir	N/Grp'd	VCM RW	90	
/dev/rdisk/c3t16d23s2	0040	02C:0	02B:D5	2-Way Mir	N/Grp'd	RW	6	
/dev/rdisk/c3t16d24s2	0041	02C:0	02A:D6	2-Way Mir	N/Grp'd	RW	6	
/dev/rdisk/c3t16d27s2	0103	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025	
/dev/rdisk/c3t16d28s2	0104	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025	
/dev/rdisk/c3t16d29s2	0105	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025	
/dev/rdisk/c3t16d30s2	0106	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025	

```
sun196/usr/sengupta> newfs /dev/rdisk/c3t16d27s2
```

After mapping and masking a reboot (or bus scan) has to be performed in order to make the devices host visible. Remember to run `symcfg discover` after the reboot so the new pdevs are recognized by SYMCLI.

Round Robin Allocation

```
sun196/> symcfg show -pool WB -thin -detail -sid 38
```

```
.....
```

Enabled Devices(2):

```
{
```

Sym Dev	Total Tracks	Alloc Tracks	Free Tracks	Full (%)	Device State
0113	16392	948	15444	5	Enabled
0114	16392	888	15504	5	Enabled
Tracks	32784	1836	30948	5	

```
}
```

```
Thin Devices(4):
```

```
{
```

Sym Dev	Total Tracks	Pool Subs (%)	Pool Allocated Tracks	Pool Allocated (%)	Pool Written Tracks	Pool Written (%)	Status
0103	16395	50	1764	5	-	-	Bound
0104	16395	50	24	0	-	-	Bound
0105	16395	50	24	0	-	-	Bound
0106	16395	50	24	0	-	-	Bound
Tracks	65580	200	1836	6	-	-	

```
}
```

© 2009 EMC Corporation. All rights reserved.

EMC Confidential – Draft Copy – Do not distribute

Virtual Provisioning with Solutions Enabler - 17



 EMC Education Services

As a result of the file system creation on one of the Thin Devices (TDEVs), 1764 tracks (roughly 110 MB) were used. This data is spread in round robin fashion over the available Data Devices (TDATs). The spreading across the backend is not exactly round robin because in going through the selection process the DA will try to find a TDAT whose local mirrors are ready and whose pointers indicates that there are free tracks available on the device. If that device is busy, the DA will traverse the linked list of data devices until it finds a data device that is ready.

Fill up Pool Before adding Datadevs

```

sun196/ dd if=/dev/zero of=/dev/rdisk/c3t16d28s2 seek=1 bs=1048576 count=950
sun196/ dd if=/dev/zero of=/dev/rdisk/c3t16d29s2 seek=1 bs=1048576 count=950
sun196/usr/sengupta/Vp> symcfg show -pool WB -detail -thin -sid 38
Symmetrix ID: 000190301138
Symmetrix ID          : 000190301138
  
```

Enabled Devices(2):

Sym Dev	Total Tracks	Alloc Tracks	Free Full Tracks (%)	Device State
0113	16392	16284	108 99	Enabled
0114	16392	15960	432 97	Enabled
Tracks	32784	32244	540 98	

Thin Devices(4):

Pool Sym Dev	Pool Total Tracks	Subs (%)	Pool Allocated Tracks (%)	Written Tracks (%)	Status
0103	16395	50	1764 5	-	Bound
0104	16395	50	15228 46	-	Bound
0105	16395	50	15228 46	-	Bound
0106	16395	50	24 0	-	Bound
Tracks	65580	200	32244 98	-	

We now write 950 MB of data to two of the TDATs causing the pool to fill up almost completely. In a production environment, the pools should be monitored and action must be taken to expand the pool as soon as a certain percent, e.g. 80% of the pool fills up.

Add Data Devices That Are Not Currently Pooled

```
sun196/usr/sengupta/Vp> symdev list -datadev -nonpooled sid 38
```

Symmetrix ID: 000190301138

Device Name		Directors			Device			
Sym	Physical	SA	:P	DA :IT	Config	Attribute	Sts	Cap (MB)
0115	Not Visible	???:?	02A:D6	2-Way	Mir	N/A (DT)	RW	1025
0116	Not Visible	???:?	01B:D4	2-Way	Mir	N/A (DT)	RW	1025
0117	Not Visible	???:?	01A:D3	2-Way	Mir	N/A (DT)	RW	1025
0118	Not Visible	???:?	02B:D1	2-Way	Mir	N/A (DT)	RW	1025
0119	Not Visible	???:?	16A:C1	2-Way	Mir	N/A (DT)	RW	1025
011A	Not Visible	???:?	01B:C1	2-Way	Mir	N/A (DT)	RW	1025

```
sun196/ symconfigure -sid 38 -cmd "add dev 115:116 to pool WB type=thin;" commit -nop
sun196/usr/sengupta/Vp> symcfg show -pool WB -detail -sid 38 -thin -all
```

Disabled Devices(2):

Sym Dev	Total Tracks	Alloc Tracks	Free Tracks	Full (%)	Device State
0115	16392	0	16392	0	Disabled
0116	16392	0	16392	0	Disabled
Tracks	32784	0	32784	0	

There are a number of points of interest in this screen output. First, the list of `-nonpooled` data devices is used as the source of devices from which to add to our existing pool. Second, we add the same number of devices as we did originally, because this ensures even distribution of data on the data devices after the currently active data devices fill up. Lastly, upon initial addition, the data devices are in `disabled` state, unless the `member state=enable` phrase was added to the `symconfigure` command line as shown earlier in this presentation

Enable Devices and Write More Data

```
sun196/ symconfigure -sid 38 -cmd "enable dev 115:116 in pool WB type=thin;" commit -nop
```

```
sun196/usr/sengupta/Vp> symcfg show -pool WB -detail -thin -sid 38
```

```
Enabled Devices(4):
```

```
{
```

Sym Dev	Total Tracks	Alloc Tracks	Free Tracks	Full (%)	Device State
0113	16392	16284	108	99	Enabled
0114	16392	15960	432	97	Enabled
0115	16392	0	16392	0	Enabled
0116	16392	0	16392	0	Enabled
Tracks	65568	32244	33324	49	

```
}
```

```
sun196/usr/sengupta/Vp> dd if=/dev/zero of=/dev/rdsd/c3t16d30s2 seek=1 bs=1048576 count=950
```

```
sun196/usr/sengupta/Vp> symcfg show -pool WB -detail -thin -sid 38
```

```
Enabled Devices(4):
```

```
{
```

Sym Dev	Total Tracks	Alloc Tracks	Free Tracks	Full (%)	Device State
0113	16392	16392	0	100	Enabled
0114	16392	16392	0	100	Enabled
0115	16392	7056	9336	43	Enabled
0116	16392	7608	8784	46	Enabled
Tracks	65568	47448	18120	72	

```
}
```

We now enable the data devices and write more data to one of the Thin devices. Notice how the new data devices are being evenly filled. If we had added just one data device, that device would have received all the writes, thus causing a performance bottleneck.

Lesson 2: Miscellaneous VP Topics

Upon completion of this lesson, you will be able to

- State how to create and bind Thin Meta Devices to a Thin Pool
- Relate how to unbind Thin Devices and free up pool space
- Indicate the process of setting subscription limit
- Describe the process of draining data devices



These are the objectives for this lesson. Please take a moment to read them.

Thin Metadevice Creation

- Created using thin devices before binding devices to pool

```
sun196/usr/sengupta/Vp> symdev list -tdev -unbound
```

```
Symmetrix ID: 000190301138
```

Device Name		Directors		Device			
Sym	Physical	SA :P	DA :IT	Config	Attribute	Sts	Cap (MB)
0107	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0108	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
0109	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025
010A	Not Visible	???:?	NA:NA	TDEV	N/Grp'd	NR	1025

```
sun196/ symconfigure -sid 38 -cmd "form meta from dev 107, config=concatenated;
add dev 108 to meta 107;" commit -nop
```

```
.....
COMMIT requesting required resources.....Obtained.
Local: COMMIT.....Done.
Terminating the configuration change session.....Done.
```

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

Thin devices can be formed into meta devices. Metadevices made from thin volumes are created using the same syntax as metadevices from regular volumes. Thin metadevices must be created before the thin devices are bound to a thin pool. If an attempt is made to form a meta device from bound metadevices, the command will fail.

Here we create a 2-member concatenated meta. As discussed earlier, with Thin devices there is no benefit to be derived from creating a striped meta, since striping is already done on the backend data devices.

symdev show on Thin Meta Device

```
sun196/usr/sengupta/Vp> symdev show 107 -sid 38
```

Device Capacity

```
{
  Cylinders      : 2186
  Tracks         : 32790
  512-byte Blocks : 4197120
  MegaBytes      : 2049
  KiloBytes      : 2098560
}
```

Device Configuration : **TDEV** (Meta Head,
Non-Exclusive Access)

Device Service State : Normal

Device Status : Not Ready (NR)
Device SA Status : N/A (N/A)

Meta Configuration : Concatenated

Meta Device Members (2) :

		BCV DATA				RDF DATA			
Sym	Cap	Std	Inv	BCV	Inv	Pair	R1	Inv	R2
Dev	(MB)	Tracks	Tracks	Tracks	Tracks	State	Tracks	Tracks	State
--> 0107	1025	-	-	-	-	N/A	-	-	N/A
0108	1025	-	-	-	-	N/A	-	-	N/A
	2049	-	-	-	-	-	-	-	-

(...continued on right column)

(...continued from left column)

Mirror Set Type : **Thin**, 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      : Thin
  Hyper Status    : Ready (RW)
  Disk [Director, Interface, TID] : [N/A, N/A, N/A]
  Disk Director Volume Number : N/A
  Hyper Number    : N/A
}
```

The output of symdev show reveals that a thin meta device looks similar to a regular meta device with differences in the Device Configuration, the Mirror Set and the Hyper Type information. There is also no backend information on the device.

Bind Meta Device to Pool

```
sun196/ symconfigure -sid 38 -cmd "bind tdev 107 to pool WB;" commit -nop
A Configuration Change operation is in progress. Please wait...
```

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

```
sun196/usr/sengupta> symcfg show -pool WB -detail -sid 38 -thin
```

```
Symmetrix ID: 000190301138
```

```
.....
Thin Devices(5):
```

```
{
```

Sym Dev	Total Tracks	Pool Subs (%)	Pool Allocated		Pool Written		Status
			Tracks	(%)	Tracks	(%)	
0103	16395	25	1764	3	-	-	Bound
0104	16395	25	15228	23	-	-	Bound
0105	16395	25	15228	23	-	-	Bound
0106	16395	25	15228	23	-	-	Bound
0107	32790	50	24	0	-	-	Bound
Tracks	98370	150	47472	72	-	-	

```
}
```

The meta can be bound to a pool and made available for use by mapping and masking the meta head (not shown here). While the initial meta creation had to be done before the thin devices were bound, later additions to the meta can be done while the metadvice is bound.

Concatenate Members to Bound Meta

```
sun196/ symconfigure -sid 38 -nop -cmd "add dev 109:10A to meta 107;" commit
A Configuration Change operation is in progress. Please wait...
```

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

```
sun196/usr/sengupta> symcfg show -pool WB -detail -sid 38 -thin
```

```
Symmetrix ID: 000190301138
```

```
.....
Thin Devices(5):
```

```
{
```

Sym Dev	Total Tracks	Pool Subs (%)	Pool Allocated		Pool Written		Status
			Tracks	(%)	Tracks	(%)	
0103	16395	25	1764	3	-	-	Bound
0104	16395	25	15228	23	-	-	Bound
0105	16395	25	15228	23	-	-	Bound
0106	16395	25	15228	23	-	-	Bound
0107	65580	100	48	0	-	-	Bound
Tracks	131160	200	47496	72	-	-	

Here we extend the meta device created earlier by two more members, thereby doubling the capacity. Since all the storage shows up under the meta head, which was already mapped and masked, the additional storage can be made available to the host using the meta with a minimum of disruption.

Unbinding Thin Devices

```
sun196/usr/sengupta> symdev list -tdev -sid 38 -range 103:10A
```

```
Symmetrix ID: 000190301138
```

Device Name		Directors		Device			
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)	
0103	/dev/rdisk/c3t16d27s2	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025
0104	/dev/rdisk/c3t16d28s2	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025
0105	/dev/rdisk/c3t16d29s2	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025
0106	/dev/rdisk/c3t16d30s2	02C:0	NA:NA	TDEV	N/Grp'd	RW	1025
0107	Not Visible	???:?	NA:NA	TDEV	N/Grp'd (M)	RW	4099
0108	Not Visible	???:?	NA:NA	TDEV	N/Grp'd (m)	RW	-
0109	Not Visible	???:?	NA:NA	TDEV	N/Grp'd (m)	RW	-
010A	Not Visible	???:?	NA:NA	TDEV	N/Grp'd (m)	RW	-

- Before unbinding thin Devices must be unmapped or set NR
- In the list above the meta device is unmapped while the regular TDEVs are mapped and RW

In order to unbind thin devices from a pool they must be unmapped or Not Ready. In the list above, the Thin devices 103-106 are mapped to FA 2C port 0, while the meta device 107 is not mapped, as indicated by the interrogation marks under the SA column.

Pool Space freed up after unbinding of TDEVs

```
sun196/ symconfigure -sid 38 -nop -cmd "unbind tdev 103:107 from pool WB;" commit
A Configuration Change operation is in progress. Please wait...
```

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

```
sun196/usr/sengupta> symcfg show -pool WB -detail -all -thin -sid 38
```

```
Symmetrix ID: 000190301138
```

```
.....
Pool State : Enabled
# of Devices in Pool : 4
# of Enabled Devices in Pool : 4
Max. Subscription Percent : None
```

```
Enabled Devices(4):
```

```
{
-----
Sym      Total   Alloc   Free Full   Device
Dev      Tracks  Tracks Tracks (%)  State
-----
0113     16392      0    16392   0  Enabled
0114     16392      0    16392   0  Enabled
0115     16392      0    16392   0  Enabled
0116     16392      0    16392   0  Enabled
-----
Tracks   65568      0    65568   0
}
```

```
No Thin Devices Bound to Device Pool WB
```

What is not shown here is the use of the command to set devices 103 to 107 Not Ready. The syntax of that command is:

```
symdev -sid 38 -range 103:107 not_ready -noprompt
```

Once Thin Devices are unbound from the pool, the space in the pool is free for use. The data on the TDEVs is lost.

Note the value of None next to Max Subscription Percent. This means that there is no limit to pool oversubscription.

Set Subscription Limit

```
sun196/usr/sengupta> symconfigure -sid 38 -nop -cmd "set pool WB, type=thin,
max_subs_percent=150;" commit
```

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

.....

The configuration change session has successfully completed.

```
sun196/usr/sengupta> symcfg show -pool WB -detail -thin -sid 38
```

Symmetrix ID: 000190301138

```
.....
Pool State                : Enabled
# of Devices in Pool      : 4
# of Enabled Devices in Pool : 4
Max. Subscription Percent : 150
```

Enabled Devices(4):

```
{
-----
Sym      Total   Alloc   Free Full   Device
Dev      Tracks  Tracks Tracks (%)  State
-----
0113     16392    0      16392  0  Enabled
0114     16392    0      16392  0  Enabled
0115     16392    0      16392  0  Enabled
0116     16392    0      16392  0  Enabled
-----
Tracks   65568    0      65568  0
}
No Thin Devices Bound to Device Pool WB
```

© 2009 EMC Corporation. All rights reserved.

EMC Confidential – Draft Copy – Do not distribute

Virtual Provisioning with Solutions Enabler - 28

 Mercer Road
EMC Education Services

A subscription limit can be set to prevent oversubscription from going beyond a specified level. This limit can be set by specifying the `max_subs_percent` value when creating a thin pool or after the pool has been created. In this command, the limit is set at 150%, meaning that the total amount of configured space for bound thin devices cannot exceed 150% of the space provided by the data devices.

Note also that the capacity of the Thin Pool is currently 5 GB, which means that a maximum of 7.5 GB will be allowed to be bound to the pool.

Enforcement of Subscription Limit

- Thin Devices 103-106 were created with 1GB capacity
- Meta device 107 has 4 GB capacity
- Devices 103:107 together exceed the 150% subscription limit

```
sun196/ symconfigure -sid 38 -nop -cmd "bind tdev 103:107 to pool WB;" prepare
```

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

```
Establishing a configuration change session.....Established.
```

```
Error occurred while Defining change number 1:  
The requested operation will exceed the pool oversubscription limit  
Device 0107 generated the failure
```

```
Terminating the configuration change session.....Done.
```

```
The configuration change session has failed.
```

If thin devices were to be bound to a pool at the time of creation, the syntax would read as follows:

```
create dev count=2, size=1GB,  
emulation=FBA, config=TDEV,  
binding to pool WB, preallocate size=1 GB;
```

Draining of Data Devices in 5874

- Thin pools can be shrunk non-disruptively
- Allocated thin device extents on DATA devices will be drained from a DATA device when it is disabled
 - Extents get written to enabled devices in the pool
- Upto 20% of the pool space can be disabled and drained at one time, i.e.
 - There have to be at least 5 devices in the pool for one of the devices to be disabled and removed from the pool

Support for the draining of data devices has been added to 5874 Enginuity for Symmetrix V-Max arrays. This allows data devices to be removed from thin pools nondisruptively.

Before removing a data device from a pool, the device must be disabled. Disabling a data device changes its state from Enabled to Draining. The concept of device draining is common to all pool types (data, DSE, and snap pools).

When a device enters a draining state, used tracks on that device are copied to the remaining, enabled devices in the pool. Therefore, data devices can be removed from a pool and repurposed without having to unbind thin devices that may have allocated extents associated with the data device.

The pool space being disabled and drained may not represent more than 20% of the available pool space.

Draining DATA Devices -1

- The pool sun180 has 5 enabled devices, each with allocated extents

```
DMX800SUN1/usr/sun1> symcfg show -pool sun180 -thin -all
Symmetrix ID: 000194900180
```

```
Symmetrix ID           : 000194900180
Pool Name              : sun180
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 Dev	Total Tracks	Used Tracks	Free Tracks	Full (%)	Device State	Session Status
0196	17244	17244	0	100	Enabled	N/A
0197	17244	17244	0	100	Enabled	N/A
0198	17244	5748	11496	33	Enabled	N/A
0199	17244	5724	11520	33	Enabled	N/A
019A	17244	5736	11508	33	Enabled	N/A
Tracks	86220	51696	34524	59		

```
}
```

Here a pool called sun180 has 5 data devices, all of which are being used.

Draining DATA Devices -2

- Disabling a DATA device causes it to drain

```
# symconfigure -sid 80 -cmd "disable dev 196 in pool sun180, type=thin;" commit
```

```
# symcfg show -pool sun180 -thin -all
```

```
.....
# of Devices in Pool      : 5
# of Enabled Devices in Pool : 4
```

```
Enabled Devices(4):
```

```
{
```

Sym Dev	Total Tracks	Used Tracks	Free Tracks	Full (%)	Device State	Session Status
0197	17244	17244	0	100	Enabled	N/A
0198	17244	7524	9720	43	Enabled	N/A
0199	17244	7500	9744	43	Enabled	N/A
019A	17244	7524	9720	43	Enabled	N/A
Tracks	68976	39792	29184	57		

```
}
```

As soon as one of the devices is disabled, it starts draining and is removed from the list of enabled devices

Draining DATA Devices -3

- Disabled DATA device will enter a draining state while its allocated extents are written to the remaining enabled devices in the pool

...

Disabled Devices(1):

{						

Sym	Total	Used	Free	Full	Device	Session
Dev	Tracks	Tracks	Tracks	(%)	State	Status

0196	17244	11916	5328	69	Draining	N/A

Tracks	17244	11916	5328	69		
}						

Device 196 is in the
process of draining

The device in question enters the “Draining” state.

Draining DATA Devices -4

- After the drain has completed, the device is now disabled

```
# symcfg show -pool sun180 -thin -all
```

```
Enabled Devices(4):
```

```
{
```

Sym Dev	Total Tracks	Used Tracks	Free Tracks	Full (%)	Device State	Session Status
0197	17244	17244	0	100	Enabled	N/A
0198	17244	11508	5736	66	Enabled	N/A
0199	17244	11436	5808	66	Enabled	N/A
019A	17244	11508	5736	66	Enabled	N/A
Tracks	68976	51696	17280	74		

```
}
```

```
Disabled Devices(1):
```

```
{
```

Sym Dev	Total Tracks	Used Tracks	Free Tracks	Full (%)	Device State	Session Status
0196	17244	0	17244	0	Disabled	N/A
Tracks	17244	0	17244	0		

```
}
```

Device 196 is now disabled

Once Draining is completed, the device is in the disabled state.

Draining DATA Devices -5

- It can now be removed from the pool

```
# symconfigure -sid 34 -cmd "remove dev 196 from pool sun180,
type=thin;" commit
```

```
# symcfg show -pool sun180 -thin -all
```

```
.....
```

Enabled Devices(4):

```
{
```

Sym Dev	Total Tracks	Used Tracks	Free Tracks	Full (%)	Device State	Session Status
0197	17244	17244	0	100	Enabled	N/A
0198	17244	11508	5736	66	Enabled	N/A
0199	17244	11436	5808	66	Enabled	N/A
019A	17244	11508	5736	66	Enabled	N/A
Tracks	68976	51696	17280	74		

```
}
```

Afterwards the device can be removed from the pool.

Module Summary

Key points covered in this module:

- Creation of Thin and Data Devices
- Addition of Data Devices to Thin Pools
- Binding Thin Devices to Thin Pools
- Creation and binding of Thin Meta Devices to a Thin Pool
- Unbinding of Thin Devices to free up pool space
- Setting of pool subscription limit
- Non-disruptive draining of Data Devices

These are the key points covered in this module. Please take a moment to review them.



Module 8: Replication with Thin Devices

Upon completion of this module you will be able to:

- Describe the interaction of Thin Devices with the following replication products:
 - TimeFinder/Snap
 - TimeFinder/Clone
 - SRDF
 - Open Replicator



These are the objectives for this module. Please take a moment to read them.

TimeFinder Snap with Thin Devices

- Thin devices can be snapped like regular devices
- Thin device sources can be snapped to virtual device targets
- VDEV targets can be made ready and updated
- VDEVs can be restored back to the source thin devices

Thin devices can be used with TimeFinder/Snap in the same way that other types of Symmetrix devices can be used. Point-in-time snap copies of thin devices can be created on virtual devices (VDEVs), which can subsequently be presented to a host. The VDEVs can also be updated and the changes on the VDEVs can be restored back to the source thin devices or to different thin devices.

Thin devices are classified as standard devices and can be added to Symmetrix device groups or put in device files.

Thin Devices to be used for Replication

```
sun196/> symcfg show -pool WB -thin -sid 38 -detail -all
```

```

Symmetrix ID: 000190301138

Symmetrix ID      : 000190301138
Pool Name         : WB
Pool Type         : Thin
Dev Emulation     : FBA
Dev Configuration : 2-Way Mir
Pool State        : Enabled
# of Devices in Pool : 2
# of Enabled Devices in Pool : 2
Max. Subscription Percent : None

Thin Devices(4):
{
-----
Sym      Pool      Pool      Pool
Dev      Total Subs  Allocated  Written
-----
0103     17250  50      12    0    -    -    Bound
0104     17250  50      12    0    -    -    Bound
0105     17250  50      12    0    -    -    Bound
0106     17250  50      12    0    -    -    Bound
-----
Tracks      69000  200      48    0    -    -
}

Enabled Devices(2):
{
-----
Sym      Total  Alloc  Free Full  Device
Dev      Tracks Tracks Tracks (%) State
-----
0113     17244    36   17208  0  Enabled
0114     17244    12   17232  0  Enabled
-----
Tracks    34488    48   34440  0
}

```

continued on right

Thin devices can be used with TimeFinder/Snap in the same way that other types of Symmetrix devices can be used. Point-in-time snap copies of thin devices can be created on virtual devices (VDEVs), which can subsequently be presented to a host. The VDEVs can also be updated and the changes on the VDEVs can be restored back to the source thin devices or to different thin devices.

Thin devices are classified as standard devices and can be added to Symmetrix device groups or put in device files.

The device pool WB shown here will be used for local replication.

Thin Device Group to be used for Snap

```
sun196/usr/sengupta/Vp> symdbg show tsnap
```

```
Group Name:  tsnap
Group Type           : REGULAR
Device Group in GNS  : No
Valid                : Yes
Symmetrix ID         : 000190301138
Group Creation Time  : Fri Feb 20 10:33:34 2009
Vendor ID            : EMC Corp
Application ID       : SYMCLI
```

Standard (STD) Devices (4):

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)
DEV001	/dev/rdisk/c3t16d27s2	0103		RW	1078
DEV002	/dev/rdisk/c3t16d28s2	0104		RW	1078
DEV003	/dev/rdisk/c3t16d29s2	0105		RW	1078
DEV004	/dev/rdisk/c3t16d30s2	0106		RW	1078

VDEV Devices Locally-associated (4):

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)
VDEV001	N/A	0061		NR	1078
VDEV002	N/A	0062		NR	1078
VDEV003	N/A	0063		NR	1078
VDEV004	N/A	0064		NR	1078

The device group shown above was created by executing the commands:

```
sun196/> symdbg create tsnap
```

```
sun196/> symld -g tsnap addall dev -range 103:106 -sid 38
```

```
sun196/> symld -g tsnap addall dev -range 61:64 -vdev
```

Snap query after Activation

```
sun196/usr/sengupta/Vp> symsnap query -g tsnap
```

```
Device Group (DG) Name: tsnap
```

```
DG's Type : REGULAR
```

```
DG's Symmetrix ID : 000190301138
```

Source Device			Target Device			State	Copy
Logical	Sym	Protected Tracks	Logical	Sym	G	Changed Tracks SRC <=> TGT	(%)
DEV001	0103	17250	VDEV001	0061	X	0 CopyOnWrite	0
DEV002	0104	17250	VDEV002	0062	X	0 CopyOnWrite	0
DEV003	0105	17250	VDEV003	0063	X	0 CopyOnWrite	0
DEV004	0106	17250	VDEV004	0064	X	0 CopyOnWrite	0
Total		-----			-----		
Track(s)		69000				0	
MB(s)		4312.5				0.0	

Legend:

(G): X = The Target device is associated with this group,
 . = The Target device is not associated with this group.

© 2009 EMC Corporation. All rights reserved.

EMC Confidential – Draft Copy – Do not distribute

Replication with Thin Devices - 5

 Mercer Road
 EMC Education Services

The commands used here were:

```
sun196/usr/sengupta/Vp> symsnap -g tsnap create -nop
```

'Create' operation execution is in progress for device group 'tsnap'. Please wait...

'Create' operation successfully executed for device group 'tsnap'.

```
sun196/usr/sengupta/Vp> symsnap -g tsnap activate -nop
```

'Activate' operation execution is in progress for device group 'tsnap'. Please wait...

'Activate' operation successfully executed for device group 'tsnap'.

Thin Devices and Clones

- Source and Target Devices must be Thin
- Thin clone targets may be accessed just as normally as regular clone targets
- Since there is no device type “Thin BCV” clone targets are addressed by
 - using the **-tgt** option with a device group
 - explicit pairing via a device group or a device file
- Space is not consumed on Clone target device until writes are directed to the source

TimeFinder/Clone can be used to create point-in-time copies of thin devices on other thin devices. Other than the requirement that both the source and target device be thin, creating clones of thin devices is exactly the same as creating clones of regular devices. Thin clones can be presented to a host and accessed just like regular devices or clone source devices. Thin clone targets can also be updated and the changes on the targets can be restored back to the source thin devices or to different thin devices.

Thin devices are classified as standard devices and can be added to Symmetrix device groups or put in device files. When using device groups, clone operations can be performed against devices by specifying the clone source and target when the clone pairs are created or clone targets can instead be added as targets by specifying the **-tgt** option when they are added to the group.

Thin Device Group to be used for cloning

```
sun196/usr/sengupta/Vp> symdbg show tclone
```

```
Group Name: tclone
Group Type                : REGULAR
Device Group in GNS       : No
Valid                     : Yes
Symmetrix ID              : 000190301138
Group Creation Time       : Fri Feb 20 11:06:24 2009
Vendor ID                 : EMC Corp
Application ID            : SYMCLI
```

Standard (STD) Devices (2):

```
{
-----
LdevName          PdevName          Sym      Cap
Dev   Att.  Sts    (MB)
-----
DEV001            /dev/rdisk/c3t16d27s2  0103      RW    1078
DEV002            /dev/rdisk/c3t16d28s2  0104      RW    1078
}
```

TGT Devices Locally-associated (2):

```
{
-----
LdevName          PdevName          Sym      Cap
Dev   Att.  Sts    (MB)
-----
TGT001            /dev/rdisk/c3t16d29s2  0105      RW    1078
TGT002            /dev/rdisk/c3t16d30s2  0106      RW    1078
}
```

The device group shown above was created by executing the commands:

```
sun196/usr/sengupta/Vp> symdbg create tclone
```

```
sun196/> symld -g tclone addall dev -range 103:104 -sid 38
```

```
sun196/> symld -g tclone addall dev -range 105:106 -tgt
```


Query output after Creation and Activation

```
sun196/usr/sengupta/Vp> symclone query -g tclone
```

```
Device Group (DG) Name: tclone
DG's Type           : REGULAR
DG's Symmetrix ID    : 000190301138
```

Source Device				Target Device			State	Copy
Logical	Sym	Protected Tracks	Modified Tracks	Logical	Sym	Modified Tracks	CGDP	SRC <=> TGT (%)
DEV001	0103	0	0	TGT001	0105	0	XXXX	PreCopy 100
DEV002	0104	0	0	TGT002	0106	0	XXXX	PreCopy 100

```
sun196/usr/sengupta/Vp> symclone query -g tclone
```

```
Device Group (DG) Name: tclone
DG's Type           : REGULAR
DG's Symmetrix ID    : 000190301138
```

Source Device				Target Device			State	Copy
Logical	Sym	Protected Tracks	Modified Tracks	Logical	Sym	Modified Tracks	CGDP	SRC <=> TGT (%)
DEV001	0103	0	0	TGT001	0105	0	XXX.	Copied 100
DEV002	0104	0	0	TGT002	0106	0	XXX.	Copied 100

The command to create the clone session:

```
sun196/> symclone -g tclone create -precopy -diff -tgt -nop
```

The command to activate the clone session:

```
sun196/> symclone -g tclone activate -tgt -nop
```

No Space used in pool WB After Clone Copy

```
sun196/> symcfg show -pool WB -thin -sid 38 -detail -all
```

Symmetrix ID: 000190301138

Thin Devices(4):

Symmetrix ID : 000190301138
 Pool Name : WB
 Pool Type : Thin
 Dev Emulation : FBA
 Dev Configuration : 2-Way Mir
 Pool State : Enabled
 # of Devices in Pool : 2
 # of Enabled Devices in Pool : 2
 Max. Subscription Percent : None

Sym Dev	Pool		Pool		Pool		Status
	Total Tracks	Subs (%)	Allocated Tracks	(%)	Written Tracks	(%)	
0103	17250	50	12	0	-	-	Bound
0104	17250	50	12	0	-	-	Bound
0105	17250	50	12	0	-	-	Bound
0106	17250	50	12	0	-	-	Bound
Tracks	69000	200	48	0	-	-	

Enabled Devices(2):

Sym Dev	Total Tracks	Alloc Tracks	Free Full Tracks	Device (%)	State
0113	17244	36	17208	0	Enabled
0114	17244	12	17232	0	Enabled
Tracks	34488	48	34440	0	

continued on right

Displaying the pool WB on Symmetrix 38 shows that the storage space allocation has not changed despite the fact that the clone copy is 100% complete.



Thin Devices and SRDF

- Source and Target devices must be thin
- Modes of RDF supported
 - SRDF/S,
 - SRDF/A
 - SRDF/Adaptive
- Space is not consumed on RDF target device until writes are directed to the source

SRDF/Synchronous, SRDF/Asynchronous, or SRDF/Adaptive Copy can be used to perform replication of thin devices to thin devices on a remote Symmetrix array. Other than the requirement that both the source and target device be thin, creating and managing thin SRDF devices is exactly the same as creating and managing SRDF using regular devices.

View of Pool WB94 on Symmetrix 94

sun197/> symcfg show -thin -pool WB94 -sid 94 -detail -all

Symmetrix ID: 000190301194

Thin Devices(4):

```
{
  -----
  Sym      Pool      Pool      Pool
  Dev      Total Subs  Allocated  Written
  -----
  0103     17250  50      12    0    -    -    Bound
  0104     17250  50      12    0    -    -    Bound
  0105     17250  50      12    0    -    -    Bound
  0106     17250  50      12    0    -    -    Bound
  -----
  Tracks   69000  200      48    0    -    -
}
```

Enabled Devices(2):

```
{
  -----
  Sym      Total  Alloc  Free Full  Device
  Dev      Tracks Tracks Tracks (%) State
  -----
  0113     17244    24   17220  0 Enabled
  0114     17244    24   17220  0 Enabled
  -----
  Tracks   34488    48   34440  0
}
```

continued on right

This is the pool WB94 on Symmetrix 94 which is RDF connected to Symmetrix 34

Prepare for RDF Pair Creation

Terminate Clone session else dynamic attribute cannot be set

```
sun196/usr/sengupta/Vp> symclone terminate -nop -g tclone -tgt
```

```
.....
'Terminate' operation successfully executed for device group
'tclone'.
```

Make Thin Devices on both Arrays Dynamic RDF capable

```
sun196/> symconfigure -sid 38 -nop -cmd "set dev 103:106 attribute=dyn_rdf;"
commit
```

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

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

```
sun197/> symconfigure -sid 94 -nop -cmd "set dev 103:106 attribute=dyn_rdf;"
commit
```

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

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

Before creating a dynamic R1-R2 pair the thin devices must be assigned the “dynamic capable” RDF attribute.

RDF Groups Connecting Symmetrix 38 and 94

```
sun196/usr/sengupta> symcfg list -rdfg all -sid 38
```

```
Symmetrix ID : 000190301138
```

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 (sec)	RA-Grp	SymmID	T	Name	Flags LPDS	Dir Cfg	Flags CSRM	Cycle time	Pri
1 (0)	10	1 (0)	000190301194	D	winrdf	XX..	F-S	-IS-	30	33
2 (1)	10	2 (1)	000190301194	D	sunrdf	XX..	F-S	-IS-	30	33
3 (2)	10	3 (2)	000190301194	D	ibmrdf	XX..	F-S	-IS-	30	33
4 (3)	10	4 (3)	000190301194	D	hprdf	XX..	F-S	-IS-	30	33
5 (4)	10	5 (4)	000190301194	D	srdfce_s	XX..	F-S	-IS-	30	33
6 (5)	10	6 (5)	000190301194	D	srdfce_a	XX..	F-S	.IS-	30	33

```
sun196/usr/sengupta/Vp> cat thinpairs
```

```
103 103
```

```
104 104
```

An examination of the RDF groups on the local Symmetrix 38 shows that there is already a group no. 2 called sunrdf that connects Symmetrix arrays 38 and 94. Had the group not existed, it would have been necessary to create a dynamic RDF group using the symrdf addgrp command. The text file thinpairs lists the devices on the source and target arrays that will later be paired up.

Create RDF Pairs

```
sun196/> symrdf createpair -est -file thinpairs -nop -sid 38 -type R1 -rdfig 2
```

An RDF 'Create Pair' operation execution is in progress for device file 'thinpairs'. Please wait...

```
Create RDF Pair in (1138,002).....Started.
Create RDF Pair in (1138,002).....Done.
Mark target device(s) in (1138,002) to refresh from source.....Started.
Devices: 0103-0104 in (1138,002)..... Marked.
Mark target device(s) in (1138,002) to refresh from source.....Done.
Mark target device(s) in (1138,002) for full copy from source....Started.
Devices: 0103-0104 in (1138,002)..... Marked.
Mark target device(s) in (1138,002) for full copy from source....Done.
Merge track tables between source and target in (1138,002).....Started.
Devices: 0103-0104 in (1138,002)..... Merged.
Merge track tables between source and target in (1138,002).....Done.
Resume RDF link(s) for device(s) in (1138,002).....Started.
Resume RDF link(s) for device(s) in (1138,002).....Done.
```

The RDF 'Create Pair' operation successfully executed for device file 'thinpairs'.

RDF pair creation is described in Chapter 2 of the Solutions Enabler SRDF manual. Using the file thinpairs shown on the last page, the RDF pairs are created.

RDF Devices Created on Source and Target

```
sun197/usr/sengupta> symdev list -range 103:104
```

Symmetrix ID: 000190301138

Device Name		Directors		Device		
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)
0103	Not Visible	02C:0 NA:NA	RDF1+TDEV	N/Grp'd	RW	1078
0104	Not Visible	02C:0 NA:NA	RDF1+TDEV	N/Grp'd	RW	1078

Symmetrix ID: 000190301194

Device Name		Directors		Device		
Sym	Physical	SA :P DA :IT	Config	Attribute	Sts	Cap (MB)
0103	/dev/rdisk/c4t17d23s2	02C:0 NA:NA	RDF2+TDEV	N/Grp'd	WD	1078
0104	/dev/rdisk/c4t17d24s2	02C:0 NA:NA	RDF2+TDEV	N/Grp'd	WD	1078

After the pair is created the R1 and R2 devices show up as Thin devices

RDF Devices synchronized

```
sun196/usr/sengupta/Vp> symrdf list -sid 38 -rdfig 2
```

Symmetrix ID: 000190301138

Local Device View

		STATUS		MODES		RDF S T A T E S				
Sym	RDF	-----		-----	R1 Inv	R2 Inv	-----			
Dev	RDev	Typ:G	SA RA LNK	MDAT	Tracks	Tracks	Dev	RDev	Pair	
0103	0103	R1:2	RW RW RW	S...	0	0	RW	WD	Synchronized	
0104	0104	R1:2	RW RW RW	S...	0	0	RW	WD	Synchronized	

```
sun197/usr/sengupta> symrdf list -rdfig 2 -sid 94
```

Symmetrix ID: 000190301194

Local Device View

		STATUS		MODES		RDF S T A T E S				
Sym	RDF	-----		-----	R1 Inv	R2 Inv	-----			
Dev	RDev	Typ:G	SA RA LNK	MDAT	Tracks	Tracks	Dev	RDev	Pair	
0103	0103	R2:2	RW WD RW	S...	0	0	WD	RW	Synchronized	
0104	0104	R2:2	RW WD RW	S...	0	0	WD	RW	Synchronized	

As shown here the R1 and R2 devices are synchronized

No Pool Space used on R2 Symmetrix 94

```
sun197/> symcfg show -thin -pool WB94 -sid 94 -detail -all
```

```

Symmetrix ID: 000190301194

Symmetrix ID      : 000190301194
Pool Name         : WB94
Pool Type        : Thin
Dev Emulation     : FBA
Dev Configuration : 2-Way Mir
Pool State        : Enabled
# of Devices in Pool : 2
# of Enabled Devices in Pool : 2
Max. Subscription Percent : None

Thin Devices(4):
{
-----
Sym      Pool      Pool      Pool
Dev      Total Sub   Allocated   Written
-----
0103     17250  50      12    0    -    -    Bound
0104     17250  50      12    0    -    -    Bound
0105     17250  50      12    0    -    -    Bound
0106     17250  50      12    0    -    -    Bound
-----
Tracks      69000  200      48    0    -    -

Enabled Devices(2):
{
-----
Sym      Total      Alloc      Free Full Device
Dev      Tracks      Tracks      Tracks (%) State
-----
0113     17244        24     17220  0 Enabled
0114     17244        24     17220  0 Enabled
-----
Tracks      34488        48     34440  0
}

```

continued on right

This is the pool WB94 on Symmetrix 94 which is RDF connected to Symmetrix 34. Note that no space has been used on the Thin pool on Symmetrix 94, despite the R2 devices being synchronized with the R1 devices.

Delete Thin RDF Pairs

```
sun196/usr/sengupta/Vp> symrdf suspend -sid 38 -file thinpairs -rdfig 2 -nop
```

An RDF 'Suspend' operation execution is in progress for device
file 'thinpairs'. Please wait...

Suspend RDF link(s) for device(s) in (1138,002).....Done.

The RDF 'Suspend' operation successfully executed for device
file 'thinpairs'.

```
sun196/usr/sengupta/Vp> symrdf deletepair -sid 38 -rdfig 2 -file thinpairs -nop
```

An RDF 'Delete Pair' operation execution is in progress for device
file 'thinpairs'. Please wait...

Delete RDF Pair in (1138,002).....Started.
Delete RDF Pair in (1138,002).....Done.

The RDF 'Delete Pair' operation successfully executed for device
file 'thinpairs'.

The RDF deletepair action dissolves the RDF relationships. This too is documented in Chapter 2 of the Solutions Enabler SRDF manual.

Open Replicator and Thin Devices

- Thin Devices can be used as control or remote devices
- Push from unused sections of Thin Devices sends 0s
- As data copy occurs, Thin Device targets become allocated i.e.
 - Control Device in an OR pull
 - Remote Device in an OR push
- If a smaller device sends data to a larger thin device only as much space will be allocated as the size of the smaller device

Open Replicator can be used to perform remote replication between thin devices or between thin and regular devices. Managing thin device replication with Open Replicator is exactly the same as managing the replication of regular devices.

Thin devices can be used as control devices for hot and cold pull and cold push Open Replicator copy operations. If a push operation is done using a thin device as the source, zeroes will be sent for any regions of the thin device that have not been allocated, or that have been allocated but have not been written to.

Open Replicator can also be used to copy data from a regular device to a thin device. If a pull or push operation is initiated from a regular device that targets a thin device, then a portion of the target thin device, equal in size to the reported size of the source volume, will become allocated.

Valid OR Targets on Symmetrix 94

```
sun196/usr/sengupta> symsan list -sanports -sid 38 -dir 2C -p 0
```

```
Symmetrix ID: 000190301138
```

Flags	Num
DIR:P I Vendor Array	LUNs Remote Port WWN

```
-----
02C:0 . EMC Symmetrix 000190301194 3 5006048AD5F10281
-----
```

```
sun196/> symsan list -sanluns -wwn 5006048AD5F10281 -sid 38 -dir 2C -p 0
```

```
Symmetrix ID: 000190301138
Remote Port WWN: 5006048AD5F10281
```

ST	A	T	Flags	Block	Capacity	LUN	Dev	LUN
DIR:P	E	ICRT	Size	(MB)	Num	Num	WWN	
02C:0	RW	...	F	512	90	0	0060	60060480000190301194533030303630
02C:0	RW	...	F	512	1078	17	0103	60060480000190301194533030313033
02C:0	RW	...	F	512	1078	18	0104	60060480000190301194533030313034

The `symsan list -sanports` command displays the WWNs that are visible across the SAN through director port 2C-0 on Symmetrix 38.

Next, the `symsan list -sanluns` command displays a list of the devices that are accessible through port 2C-0 and the WWN 5006048AD5F1028. These are the same devices that were masked on the previous slide, namely devices 103 and 104 on Symmetrix 94

OR Device File with 1 TDEV and 1GK as control devices

```
sun196/usr/sengupta/Vp> symdev list -range 40:40 -sid 38
```

```
Symmetrix ID: 000190301138
```

Device Name		Directors		Device		
Sym	Physical	SA	:P DA :IT Config	Attribute	Sts	Cap (MB)
0040	/dev/rdisk/c3t16d23s2	02C:0	02B:D5	2-Way Mir	N/Grp'd	RW 6

```
sun196/usr/sengupta/Vp> cat orpairs
```

```
# Control Remote
symdev=38:103 symdev=94:103
symdev=38:40 symdev=94:104
```

- Copy will occur from devices on Symmetrix 38 to Symmetrix 94
- Device 103 is a Thin Device on both Symmetrix arrays
- GK Device 40 chosen to show effect of small to large replication

The Open Replicator device file contains one 1GB Thin device and a small gatekeeper device with a capacity of 6 MB.

View of Target Pool WB94 Before OR Data Transfer

sun197/> symcfg show -thin -pool WB94 -sid 94 -detail -all

Symmetrix ID: 000190301194

Thin Devices(4):

Symmetrix ID : 000190301194
 Pool Name : WB94
 Pool Type : Thin
 Dev Emulation : FBA
 Dev Configuration : 2-Way Mir
 Pool State : Enabled
 # of Devices in Pool : 2
 # of Enabled Devices in Pool : 2
 Max. Subscription Percent : None

Sym Dev	Pool		Pool		Pool		Status
	Total Tracks	Subs (%)	Allocated Tracks	(%)	Written Tracks	(%)	
0103	17250	50	12	0	-	-	Bound
0104	17250	50	12	0	-	-	Bound
0105	17250	50	12	0	-	-	Bound
0106	17250	50	12	0	-	-	Bound
Tracks	69000	200	48	0	-	-	

Enabled Devices(2):

Sym Dev	Total Tracks	Alloc Tracks	Free Full Tracks	Device (%)	State
0113	17244	24	17220	0	Enabled
0114	17244	24	17220	0	Enabled
Tracks	34488	48	34440	0	

continued on right

This is the pool WB94 on Symmetrix 94 which is RDF connected to Symmetrix 34

Create OR Session

```
sun196/usr/sengupta/Vp> symrcopy create -copy -hot -push -nop -pace 0 -file orpairs
```

```
sun196/usr/sengupta/Vp> symrcopy query -file orpairs
```

```
Device File Name      : orpairs
```

Control Device	Remote Device	Flags	Status	Done
SID:symdev	Protected Tracks	Identification	RI CDSHU CTL <=> REM	(%)
000190301138:0103	17250	000190301194:0103	SD X.XX. Created	N/A
000190301138:0040	90	000190301194:0104	SD X.XX. Created	N/A

- **-copy** option starts background data copy after activation of hot push session
- Protected tracks are actual sizes of the source devices
- Pace set to 0 to make data transfer as fast as possible

The Open Replicator session is created as shown above with the **-copy** option. This option starts the background copy of data immediately after session activation. This is a hot push session, so the remote devices on Symmetrix 94 will go through storage space allocation as the data copy starts right after activation.

The protected tracks of 17250 and 90 reflect the reported sizes of the control devices on the local Symmetrix. Setting the pace to zero speeds up the data transfer.

Activate Open Replicator Session

```
sun196/usr/sengupta/Vp> symrcopy activate -nop -file orpairs
sun196/usr/sengupta/Vp> symrcopy query -file orpairs
```

Device File Name : orpairs

Control Device	Remote Device	Flags	Status	Done

SID:symdev	Protected Tracks Identification	RI CDSHU	CTL <=> REM	(%)

000190301138:0103	0 000190301194:0103	SD X.XX.	Copied	100
000190301138:0040	0 000190301194:0104	SD X.XX.	Copied	100

- No change happens in pool WB94 after session creation hence no list output shown after session creation
- After activation background data copy finishes

Since no space is allocated in the pool WB94 when the Open Replicator session is created, the listing of the pool has not been shown after session creation. After the session is activated, the data from control to remote is copied in the background because of the `-copy` option in the command line.

Pool WB94 After Completion of OR Data Transfer

```
sun197/> symcfg show -thin -pool WB94 -sid 94 -detail -all
```

Symmetrix ID: 000190301194

Symmetrix ID : 000190301194
 Pool Name : WB94
 Pool Type : Thin
 Dev Emulation : FBA
 Dev Configuration : 2-Way Mir
 Pool State : Enabled
 # of Devices in Pool : 2
 # of Enabled Devices in Pool : 2
 Max. Subscription Percent : None

Thin Devices(4):

{

-

Sym Dev	Pool Total Tracks	Subs (%)	Pool Allocated Tracks	Pool Allocated (%)	Pool Written Tracks	Pool Written (%)	Status
0103	17250	50	17256	50	-	-	Bound
0104	17250	50	108	0	-	-	Bound
0105	17250	50	12	0	-	-	Bound
0106	17250	50	12	0	-	-	Bound
Tracks	69000	200	17388	50	-	-	

Enabled Devices(2):

{

Sym Dev	Total Tracks	Alloc Tracks	Free Tracks	Full Tracks (%)	Device State
0113	17244	8688	8556	50	Enabled
0114	17244	8700	8544	50	Enabled
Tracks	34488	17388	17100	50	

Allocated tracks rounded up to nearest multiple of 12

continued on right

A look at the WB94 pool shows that device 103 is fully allocated with 17256 tracks and device 104, which received data from the small device 40, has 108 tracks allocated.

Note that on the data devices, the available tracks have been rounded down from 17250, which is the actual size of the device, to the closest multiple of 12. On the Thin devices, space allocated has been rounded up to the closest multiple of 12.

Module Summary

Key points covered in this module:

- Thin Devices with TimeFinder/Snap
- Thin Devices with TimeFinder/Clone
- Thin Devices with SRDF
- Thin Devices with Open Replicator

These are the key points covered in this module. Please take a moment to review them.

Module 9: Monitoring Thin Pools

Upon completion of this module you will be able to:

- Describe the monitoring of Thin pools with SYMCLI
- Explain the monitoring of Thin pools with SMC
- Describe the monitoring of Thin pools with the event daemon



These are the objectives for this module. Please take a moment to read them.

Monitor Thin Pool Space with SYMCLI

- **symcfg monitor** command starts monitoring
- User created action script can be executed when a threshold is reached

- Example:

```
symcfg -action ./action monitor -i 300 -pool WB94 -thin -sid 94 -percent 60
```

- **-norepeat** option available for one time script execution

One way to monitor thin pool space is to run the `symcfg monitor` command with the `-i` option. This command runs every “i” seconds and permits the specification of a user defined action script once a pool threshold has been reached. The action script can be set to execute each time the condition is encountered, e.g. a 60% pool full condition or only once. In the latter event the `-norepeat` option causes the action script to execute only once.

The action script can perform any task that the storage administrator chooses to perform ranging from sending an e-mail or writing to a log file up to executing a task that adds additional enabled data devices to the pool in question.

Alert Severities

Severity of Alerts Related to

- Pool free space
- Thin Device allocation
- Thin Device usage

Percent Full	Severity
60	Warning
65	Minor Warning
70	Major Warning
80	Critical
100	Fatal

There are three types and levels of alerts that can be triggered by SMC. Alert severities will be warning, critical, or fatal depending on the type of alert and the threshold that is being equaled or exceeded.

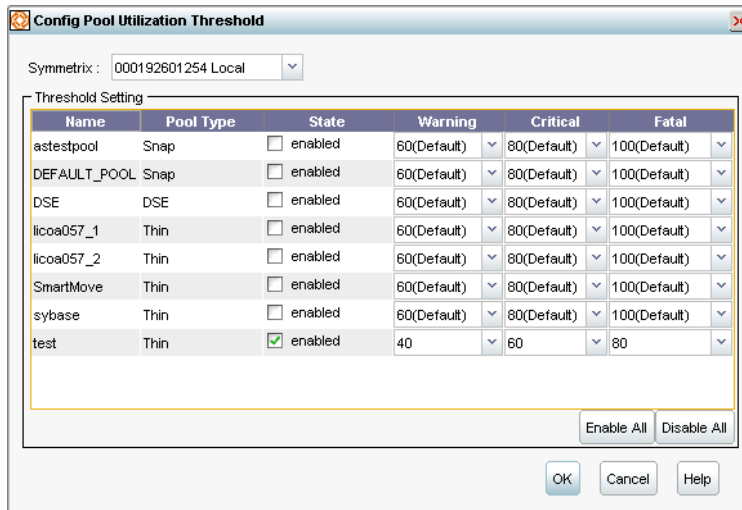
Alerts for pool or thin device usage or allocation (Device Pool Free Space, Thin Device Allocation, and Thin Device Usage) will trigger when the following thresholds are met or exceeded:

- 60% - issue a warning alert
- 65% - issue a warning alert
- 70% - issue a warning alert
- 80% - issue a critical alert
- 100% - issue a fatal alert

These are the default values coded in SMC and. Alerts that have not been cleared by the user can be viewed by clicking the Alerts button. They can alternately be viewed by clicking the New Alerts button at the upper right of the browser window. This button will also show the number of new alerts (alerts that have not been acknowledged by the user).

Pool Usage Thresholds Set by user

- Non-default pool usage thresholds can be set by user



The dialog box titled "Config Pool Utilization Threshold" shows a Symmetrix selection dropdown set to "000192601254 Local". Below is a "Threshold Setting" table with columns: Name, Pool Type, State, Warning, Critical, and Fatal. The table lists several pools, most with default thresholds (60, 80, 100). The "test" pool is highlighted with a yellow border and has custom thresholds of 40, 60, and 80. At the bottom right of the table are "Enable All" and "Disable All" buttons. At the bottom of the dialog are "OK", "Cancel", and "Help" buttons.

Name	Pool Type	State	Warning	Critical	Fatal
astestpool	Snap	☐ enabled	60(Default)	80(Default)	100(Default)
DEFAULT_POOL	Snap	☐ enabled	60(Default)	80(Default)	100(Default)
DSE	DSE	☐ enabled	60(Default)	80(Default)	100(Default)
licoa057_1	Thin	☐ enabled	60(Default)	80(Default)	100(Default)
licoa057_2	Thin	☐ enabled	60(Default)	80(Default)	100(Default)
SmartMove	Thin	☐ enabled	60(Default)	80(Default)	100(Default)
sybase	Thin	☐ enabled	60(Default)	80(Default)	100(Default)
test	Thin	☒ enabled	40	60	80

In version of 7.0 of SMC Device Pool Free Space can be customized by the user to allow specific pools to be monitored for user defined thresholds.

The dialog box for configuring pool thresholds can be found in the Tasks screen by clicking Config Pool Utilization Threshold... under Administration:

Sample Alerts

Alerts - 000190301194 Local					
State	Severity	Object	Message	Created	Last Modified
		0104	Thin Device is now 100 percent allocated. Object is: 000190301194:0104	Mon Feb 23 17:30:27 EST 2009	
		WB94	Data Pool utilization is now 100 percent. Object is: 000190301194:WB94	Mon Feb 23 17:30:27 EST 2009	
		0103	Thin Device is now 60 percent allocated. Object is: 000190301194:0103	Mon Feb 23 17:29:27 EST 2009	
		0104	Thin Device is now 60 percent allocated. Object is: 000190301194:0104	Mon Feb 23 17:29:27 EST 2009	
		WB94	Data Pool utilization is now 60 percent. Object is: 000190301194:WB94	Mon Feb 23 17:29:27 EST 2009	
		0103	Thin Device is now 0 percent allocated. Object is: 000190301194:0103	Mon Feb 23 17:28:26 EST 2009	
		0104	Thin Device is now 0 percent allocated. Object is: 000190301194:0104	Mon Feb 23 17:28:26 EST 2009	
		WB94	Data Pool utilization is now 0 percent. Object is: 000190301194:WB94	Mon Feb 23 17:28:26 EST 2009	
		0103	Thin Device is now 100 percent allocated. Object is: 000190301194:0103	Mon Feb 23 17:24:24 EST 2009	
		0104	Thin Device is now 100 percent allocated. Object is: 000190301194:0104	Mon Feb 23 17:24:24 EST 2009	
		WB94	Data Pool utilization is now 100 percent. Object is: 000190301194:WB94	Mon Feb 23 17:24:24 EST 2009	
		0103	Thin Device is now 0 percent allocated. Object is: 000190301194:0103	Mon Feb 23 17:18:21 EST 2009	
		0104	Thin Device is now 0 percent allocated. Object is: 000190301194:0104	Mon Feb 23 17:18:21 EST 2009	
		WB94	Data Pool utilization is now 0 percent. Object is: 000190301194:WB94	Mon Feb 23 17:18:21 EST 2009	
		0103	Thin Device is now 100 percent allocated. Object is: 000190301194:0103	Mon Feb 23 16:57:10 EST 2009	
		0104	Thin Device is now 100 percent allocated. Object is: 000190301194:0104	Mon Feb 23 16:57:10 EST 2009	
		WB94	Data Pool utilization is now 100 percent. Object is: 000190301194:WB94	Mon Feb 23 16:57:10 EST 2009	
		0103	Thin Device is now 65 percent allocated. Object is: 000190301194:0103	Mon Feb 23 16:56:09 EST 2009	
		0104	Thin Device is now 65 percent allocated. Object is: 000190301194:0104	Mon Feb 23 16:56:09 EST 2009	
		WB94	Data Pool utilization is now 65 percent. Object is: 000190301194:WB94	Mon Feb 23 16:56:09 EST 2009	
		0103	Thin Device is now 0 percent allocated. Object is: 000190301194:0103	Mon Feb 23 16:36:59 EST 2009	
		000190301194 Local	SIM message initiated to a remote SRDF attached array.	Mon Feb 23 13:58:04 EST 2009	
		000190301194 Local	SIM message initiated to a remote SRDF attached array.	Mon Feb 23 13:54:11 EST 2009	
		0103	Thin Device is now 100 percent allocated. Object is: 000190301194:0103	Mon Feb 23 10:44:54 EST 2009	
		0103	Device configuration has changed Object is: 000190301194:0103	Mon Feb 23 10:27:45 EST 2009	
		0104	Device configuration has changed Object is: 000190301194:0104	Mon Feb 23 10:27:45 EST 2009	
		0103	Device configuration has changed Object is: 000190301194:0103	Mon Feb 23 10:24:44 EST 2009	
		0103	Thin Device is now 0 percent allocated. Object is: 000190301194:0103	Mon Feb 23 10:24:44 EST 2009	

In the sample shown above the alert's severity, as displayed by SMC are:

- (1) Fatal
- (2) Critical
- (3) Warning
- (4) Information
- (5) Normal

Event Daemon Overview

- Optional Solutions Enabler daemon
 - storevntd
- Monitors Symmetrix operations by detecting and reporting events as they happen
 - Continually collects Symmetrix event information in real-time
 - Filters events by severity and type
- Reports all events related to Symmetrix arrays
- Supported on UNIX, Linux and Windows platforms
- Requires Solutions Enabler 6.2 or higher
 - Included with Solutions Enabler BASE license

The event daemon (**storevntd**) enables the monitoring of Symmetrix operations by detecting and reporting events as they happen. The event daemon continually collects Symmetrix event information in real-time, filters the events by severity and type, and responds by alerting client event applications and/or logging events to specified targets.

When using the daemon with a client event application (for example, the Symmetrix Management Console), the application registers with the event daemon, specifying the events in which it is interested. When used in this manner, the daemon will automatically start when the client application requests its services.

When configuring the daemon to log events, it is possible to set it up to log the events to a remote Syslog server, the UNIX Syslog, the Windows Event log, SNMP, and/or a file on disk. The event daemon is also supported on z/OS. The daemon should typically be configured to automatically start at system boot.

Event Daemon Features

- Started, stopped and managed via stordaeomon command
- Can be set to start on system boot
- Event daemon restarted by watchdog daemon (storwatchd) in the event it fails or is stopped

By default, the event daemon will automatically start the first time a Solutions Enabler application requires its services.

The daemon can be manually started using the stordaeomon command.

```
stordaeomon start storevntd
```

Alternatively it can be automatically started every time the local host is booted using the following command:

```
stordaeomon install storevntd -autostart
```

Documentation on Event Daemon

- Installation Guide
- /var/symapi/config/daemon_options
- **stord daemon action storevntd -cmd help**
- Some options shown here are undocumented
 - **thresh_critical, thresh_major, thresh_warn, etc.**
 - **sev=<info | warning | minor | major | critical | fatal>**

The event daemon is documented in the Solutions Enabler Installation Guide. The amount of information is somewhat limited. The most informative source is the daemon_options file, which has examples of how to set up the options. Some of the options shown here are undocumented and was taken from Engineering documentation.

Logging Options

- Reports events in two ways
 - Alerts client event applications
 - Logs events to one of four specified targets
 - UNIX syslog
 - Windows Event Log
 - SNMP
 - Flat file
- Client event applications register with the daemon
 - Specifies events of interest
 - Daemon automatically started when client application requests services
 - e.g. SMC, Control Center

To manually perform monitoring with the event daemon, a user needs to specify a target for the daemon's messages. The targets can be one or a combination of:

file Events are written to a file on disk.

snmp Events are mapped into SNMP traps.

system Events are written to the the local host's syslog services on Unix. The syslog's configuration settings determine which log file the message is written to. On Windows the messages are sent to the Event Log.

syslog Events are sent directly to a remote syslog server, bypassing any local syslog service.

Special applications such as SMC or Control Center have been written to take advantage of the event daemon. These applications request the information they need and the event daemon supplies it.

Reporting of Symmetrix Events

- Entries in the Daemon Options file

```
storevntd:log_symmetrix_events = sid=000190101906,events,disk,device;
storevntd:log_symmetrix_events = sid=000190101906,1208,1211;
```

- sid: Symmetrix ID of array(s) to be monitored
- Events can be a family of related events or a specific error
 - Full list of categories and error codes available in Solutions Enabler Installation Guide
- The `/var/symapi/config/daemon_options` file has extensive documentation on the even daemon

The list of events to be logged are specified in the daemon options file. The event daemon can report on a large number of categories of events which are documented in the Solutions Enabler Installation Guide. The list of individual event codes are documented in a table that covers 12 pages in the Appendix of the Guide.

Certain events correspond to numerical quantities of some sort. A threshold is associated with each severity level, and an event is generated at that severity when the event's value exceeds the associated threshold. These fields can be used to override the default threshold values controlling when an event is delivered.

One example of this is the event that indicates the percentage of space used for a device pool. These fields can be set to control when events are to be generated, e.g.: `thresh_critical=96`, `thresh_major=80`, `thresh_warn=60`, `thresh_info=40`.

An easy way to find the set of supported events and categories is by querying a running event daemon.

Load the Symmetrix detector:

```
stordaemon action storevntd -cmd load_plugin Symmetrix
```

List the event categories that it supports:

```
stordaemon action storevntd -cmd list -categories
```

SNMP Traps

- SNMP events logged via SNMP traps to a remote SNMP management server
 - Encoded according to the Fibre Channel Alliance MIB (version 3.0)
- **snmp_trap_client_registration**
 - Host name or IP address.
 - Port at this host that the traps should be sent to.
 - The trap sending filter levels as defined in the FC-management (Fibre Channel) MIB

The event daemon provides the necessary SNMP MIB support and trap generation services required to monitor the status of Symmetrix storage environments from third-party enterprise management frameworks. The event daemon includes a loadable SNMP library which, once enabled and configured in the `daemon_options` file, acts as a self contained SNMP agent. It is responsible for maintaining internal Fibre Alliance MIB (V3.0) tables, responding to SNMP browse requests, and generating traps in response to events.

For an application to receive SNMP trap information from the event daemon, it must be specified as a trap target. The application's IP address, the port on which the application will be listening for the trap and the filter that determines the highest severity level for which traps will be sent.

The possible values range from 1 through 10, where:

- 1 = Unknown
- 2 = Emergency
- 3 = Alert
- 4 = Critical
- 5 = Error
- 6 = Warning
- 7 = Notify
- 8 = Info
- 9 = Debug
- 10 = Mark (all messages logged)

Starting the Event Daemon

- To manually start daemon, issue command:
`# stordaeon start storevntd`
- To start the daemon automatically at system boot
`# stordaeon install storevntd -autostart`
- When daemon starts, it reads options from `daemon_options` and prints what it read in the event daemon log
`/var/symapi/log/stordevntd.log0` or `stordevntd.log1`
- Events are recorded in the event log whose default name is:
`/var/symapi/log/events.log0` or `events.log1`
- To have the event daemon re-read the daemon options file
`# stordaeon action storevntd -cmd reload`

The event daemon can be started explicitly by the user or set to start automatically every time the system boots. When the daemon starts it prints an entry in the event daemon log `storevntd.log0` or `storevntd.log1` file. These are circular logs that grow to 1 MB size before they get overwritten. The event daemon log notes the options that it read from the daemon options file at the time it started.

A record of events is noted in the events log, which is also a circular file called `events.log0` or `events.log1`. The file can be given a different name if the user chooses to specify it in the daemon options file.

Event Codes for VP Monitoring

Excerpt from Installation Guide (Appendix B)

Code	Category Event Description
1206	Status - Pool state has changed to [Not Present Unknown Online Offline Write Disabled Failed].
1207	Status - Pool configuration has changed.
1208	Status - Pool utilization is now nn%
1212	Status - Virtual device is now N percent allocated.
1213	Status - Virtual device is now N percent used

The table is excerpted from the Appendix of the installation guide and contains the event codes related to Thin provisioning.

Sample Entries for Monitoring Device Pools

```
storevntd:symm_poll_interval = 20 1
storevntd:log_event_targets = file 2
storevntd:log_symmetrix_events = sid=000190301194,1206,1207;\ 3
    sid=000190301194,1208,thresh_critical=80,thresh_major=65,thresh_warn=50,thresh_
4 info=40;\
    sid=000190301194,1212,thresh_critical=80,thresh_major=65,thresh_warn=50,thresh_
    info=40;\
    sid=000190301194,1213,thresh_critical=80,thresh_major=65,thresh_warn=50,thresh_
    info=40;
```

1 Polling Interval set to 20 seconds (default is 60 seconds)

2 Events will be logged to a file (default name is events.log?)

3 Pool state and configuration changes will be logged

4 For thin pool utilization, thin device allocation and thin device usage, 4 messages will be generated between 40 and 80%

This is an example that uses undocumented threshold values to specify information and different warning levels.

The options in the daemon options file are read by the event daemon at the time of startup. While the event daemon is running, these options stay in force. To dynamically alter the options, the simplest way is to edit the options file and issue the command:

```
# stordaeomon action storevntd -cmd reload
```

Another way is to use the command

```
# stordaeomon setvar storevntd -name var=Value
```

Entries in the Event Log - 1

- **Disable Data Pool**

```
[fmt=evt] [evtid=1206] [date=2009-02-26T14:22:50] [symid=000190301194]  
[TPDataPool=WB94] [sev=major] = Data Pool state has changed to Offline.
```

```
[fmt=evt] [evtid=1207] [date=2009-02-26T14:22:50] [symid=000190301194]  
[TPDataPool=WB94] [sev=info] = Data Pool configuration has changed.
```

- **Enable Data Pool**

```
[fmt=evt] [evtid=1206] [date=2009-02-26T14:24:10] [symid=000190301194]  
[TPDataPool=WB94] [sev=normal] = Data Pool state has changed to Online.
```

```
[fmt=evt] [evtid=1207] [date=2009-02-26T14:24:10] [symid=000190301194]  
[TPDataPool=WB94] [sev=info] = Data Pool configuration has changed.
```

The two entries in the event log pertain to the enabling and disabling of the Thin data device pools.

Entries in the Event Log - 2

```
[fmt=evt] [evtid=1212] [date=2009-02-26T11:17:50] [symid=000190301194] [Device=0103]
[sev=info] = Thin Device is now 40 percent allocated.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:17:50] [symid=000190301194] [Device=0104]
[sev=info] = Thin Device is now 40 percent allocated.
[fmt=evt] [evtid=1208] [date=2009-02-26T11:17:50] [symid=000190301194] [TPDataPool=WB94]
[sev=info] = Data Pool utilization is now 40 percent.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:19:31] [symid=000190301194] [Device=0103]
[sev=warning] = Thin Device is now 50 percent allocated.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:19:31] [symid=000190301194] [Device=0104]
[sev=warning] = Thin Device is now 50 percent allocated.
[fmt=evt] [evtid=1208] [date=2009-02-26T11:19:31] [symid=000190301194] [TPDataPool=WB94]
[sev=warning] = Data Pool utilization is now 50 percent.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:21:53] [symid=000190301194] [Device=0103]
[sev=major] = Thin Device is now 65 percent allocated.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:21:53] [symid=000190301194] [Device=0104]
[sev=major] = Thin Device is now 65 percent allocated.
[fmt=evt] [evtid=1208] [date=2009-02-26T11:21:53] [symid=000190301194] [TPDataPool=WB94]
[sev=major] = Data Pool utilization is now 65 percent.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:24:35] [symid=000190301194] [Device=0103]
[sev=critical] = Thin Device is now 80 percent allocated.
[fmt=evt] [evtid=1212] [date=2009-02-26T11:24:35] [symid=000190301194] [Device=0104]
[sev=critical] = Thin Device is now 80 percent allocated.
[fmt=evt] [evtid=1208] [date=2009-02-26T11:24:35] [symid=000190301194] [TPDataPool=WB94]
[sev=critical] = Data Pool utilization is now 80 percent.
```

The four levels of messages on this page pertain to the four severities specified in the options file. The messages are issued once for each Thin device bound to the pool and once for the pool as a whole.

Second Options Sample and Daemon Log Entry

```
storevntd:symm_poll_interval = 10 1
storevntd:symm_sync_frequency = 1 2
storevntd:log_event_targets = file 3
storevntd:log_symmetrix_events = \
    sid=000194900180,1208, sev=warning;\ 4
    sid=000194900180,1212, sev=warning;
```

daemon_options

All messages of severity warning or higher will be recorded in the event log

```
-----
[4810      Listener] Jun-23 12:55:46.989 : storevntd.log0 or storevntd.log1
=====
[4810      Listener] Jun-23 12:55:46.989 : storevntd Starting, Version: V7.0-915 (0.0)
[4810      Listener] Jun-23 12:55:47.028 : Event Channel from remote hosts will not be secure
[4810      Listener] Jun-23 12:55:47.028 : Option setting: log_event_targets = file 3
[4810      Listener] Jun-23 12:55:47.029 : Option setting: log_symmetrix_events = 4
sid=000194900180,1208, sev=warning;    sid=000194900180,1212, sev=warning;
[4810      Listener] Jun-23 12:55:47.068 : Option setting: symm_poll_interval = 10 1
[4810      Listener] Jun-23 12:55:47.068 : Option setting: symm_sync_frequency = 1 2
[4810      Listener] Jun-23 12:55:47.069 : Loaded Event PlugIn SymmetrixPlugin
[4810      Listener] Jun-23 12:55:48.710 : storevntd Daemon : up and running
```

This is a simpler example of how options can be set up in the daemon options file. The `sev=warning` is an undocumented option and specifies that every 1208 (thin pool) and 1212 (thin device) event that generates a warning or something more serious than a warning should be logged. Other permitted values are:

`sev=normal` `sev=info` `sev=minor` `sev=major` `sev=critical`

An excerpt from the event daemon log is shown here. This log holds the start and stop history of the event daemon. It also records the options it read from the daemon options file when it started. If the options are reloaded using the `stordaeomon` action `storevntd -cmd reload` command, that is recorded in the event daemon log as well.

The `storevntd:symm_sync_frequency` is an undocumented option. Setting it to 1 causes the event log to be updated every 30 seconds as opposed to every 2.5 minutes. This option is not really necessary in real life environments but it is useful for demonstration or lab environments.

Entries in Event Log - 3

[fmt=evt] [evtid=1212] [date=2009-06-23T13:05:29] [symid=000194900180] [Device=0191] [sev=warning] = Thin Device is now 60 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:06:00] [symid=000194900180] [Device=0191] [sev=major] = Thin Device is now 70 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:06:30] [symid=000194900180] [Device=0191] [sev=critical] = Thin Device is now 80 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:07:31] [symid=000194900180] [Device=0191] [sev=fatal] = Thin Device is now 100 percent allocated.

[fmt=evt] [evtid=1208] [date=2009-06-23T13:09:03] [symid=000194900180] [TPDataPool=sun180] [sev=warning] = Data Pool utilization is now 60 percent.

[fmt=evt] [evtid=1208] [date=2009-06-23T13:09:34] [symid=000194900180] [TPDataPool=sun180] [sev=minor] = Data Pool utilization is now 65 percent.

[fmt=evt] [evtid=1208] [date=2009-06-23T13:10:04] [symid=000194900180] [TPDataPool=sun180] [sev=major] = Data Pool utilization is now 70 percent.

[fmt=evt] [evtid=1208] [date=2009-06-23T13:11:06] [symid=000194900180] [TPDataPool=sun180] [sev=critical] = Data Pool utilization is now 80 percent.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:11:36] [symid=000194900180] [Device=0192] [sev=minor] = Thin Device is now 65 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:12:07] [symid=000194900180] [Device=0192] [sev=major] = Thin Device is now 75 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:12:37] [symid=000194900180] [Device=0192] [sev=critical] = Thin Device is now 85 percent allocated.

[fmt=evt] [evtid=1212] [date=2009-06-23T13:13:38] [symid=000194900180] [Device=0192] [sev=fatal] = Thin Device is now 100 percent allocated.

[fmt=evt] [evtid=1208] [date=2009-06-23T13:13:38] [symid=000194900180] [TPDataPool=sun180] [sev=fatal] = Data Pool utilization is now 100 percent.

Default Thresholds

60 Warning 65 Minor Warning 70 Major Warning 80 Critical 100 Fatal

As a result of the previous options, these entries were recorded in the event log as two thin devices and the thin pool got filled up. Warnings are issued whenever the thresholds are exceeded.

Module Summary

Key points covered in this module:

- Monitoring of Thin pools with SYMCLI
- Monitoring of Thin pools with SMC
- Monitoring of Thin pools with the event daemon

These are the key points covered in this module. Please take a moment to review them.

Module 10: Configuration with Symmetrix Management Console

Upon completion of this module, you will be able use the Symmetrix Management Console to:

- Set Symmetrix and Symmetrix Port attributes
- Configure Symmetrix devices
- Manage Device pools
- Allocate storage to hosts with Autoprovisioning groups
- Configure SRDF groups and RDF pairs
- Perform LUN migration



The objectives for this module are shown here. Please take a moment to read them.

Please note that in this module only the “how to” using SMC is discussed. In the previous modules we have covered all these configuration activities with SYMCLI including the theory, caveats and best practices.

Lesson 1 - Set Symmetrix and Port Attributes

Upon completion of this lesson, you will be able to use SMC to:

- Set Symmetrix attributes
- Monitor Symmetrix configuration sessions
- Set port attributes



Symmetrix Properties – Current Configuration

Properties - 000194900180 Local	
Product Model	VMAX-1SE
Microcode Version (Number)	5874.122
Microcode Date/Patch Date	04-28-2009/04-28-2009
Used Capacity (GB / MB / Cyl)	832 / 851988 / 903414
Free Capacity (GB / MB / Cyl)	10072 / 10313612 / 10823306
Capacity (GB / MB / Cyl)	10904 / 11165600 / 11726720
Actual Capacity (GB / MB / Cyl)	10904 / 11165600 / 11726720
Cache Size (MB)	12288
Number of Front End Directors	4
Number of Back End Directors	8
Number of RDF Directors	4
Number of Configured (Symm) Devices	401
Number of Visible (Host) Devices	64
Number of Configured Actual Disks	40
Number of Configured Spare Disks	8
Number of Unconfigured Disks	0
Maximum Number of Hypers per Disk	512
Device Masking (VCM) Config State	N/A
VCMDB Access Restricted State	N/A
Device Masking (ACLX) Config	Enabled
Access Control Configuration State	Enabled
Configuration Change State	Enabled
Disk Group Assignments	In Use
Parity RAID Configuration	N/A
RAID Configuration	RAID-5 (3+1)
PAV Mode	DynamicStandardPAV
PAV Alias Limit	31
SDDF Configuration State	Enabled
WORM Configuration Level	N/A
WORM Characteristics	N/A

Creation of new devices requires
adequate Free capacity

The properties view of a Symmetrix array is very detailed and shows the current configuration settings.

One can determine the Total, Free and Used capacity. The creation of new Symmetrix devices requires adequate free capacity.

Symmetrix Properties – Current Configuration ...

Switched RDF Configuration State	Enabled
Concurrent RDF Configuration State	Enabled
Dynamic RDF Configuration State	Enabled
Concurrent Dynamic RDF Configuration	Enabled
RDF Data Mobility Configuration State	Disabled
SRDF/A Maximum Host Throttle (Secs)	0
SRDF/A Maximum Cache Usage (Percent)	94
Multi LRU Device Assignment	NONE
# of Available Cache Slots	161616
Max # of DA Write Pending Slots	0
Max # of System Write Pending Slots	129419
Max # of Device Write Pending Slots	6470
Symmetrix Last IPL Time (Cold)	2009-05-22 13:03:00
Symmetrix Last Fast IPL Time (Hot)	2006-12-31 19:00:00
Symmetrix Alerts (Enabled/Disabled)	23/0
Symmetrix Priority Controls	Disabled
Hot Swap Policy	Permanent
Symmetrix Disk Library	Disabled
FBA Geometry Emulation	Native
3 Dynamic Mirrors	Enabled
Cache Partitioning	Disabled
Auto Meta	Disabled
Minimum Auto Meta Size (Cyl)	262669
Auto Meta Member Size (Cyl)	0
Auto Meta Configuration	N/A

SRDF Configuration state – Cannot be changed by SMC for V-Max

Attribute can be changed by SMC

Attribute can be changed by SMC

Auto Meta – Attributes can be changed by SMC

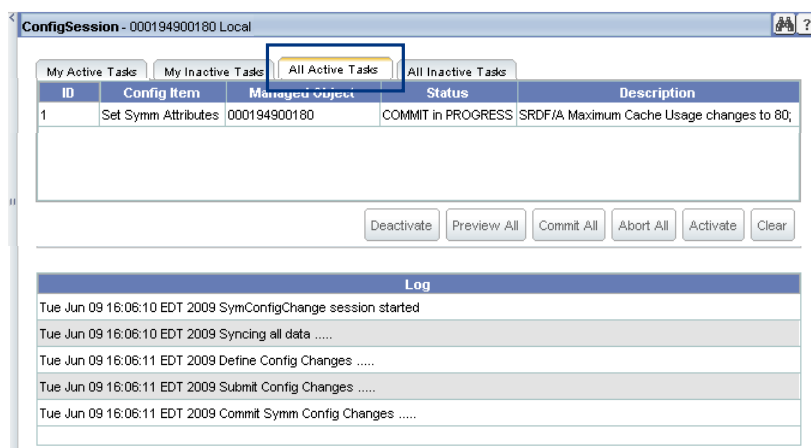
This is a continuation of the properties view of a Symmetrix. On a Symmetrix V-Max array Switched RDF, Dynamic RDF, Concurrent RDF and Concurrent Dynamic RDF are enabled by default and cannot be changed by SMC. SRDF/A maximum host throttle time, SRDF/A maximum cache usage, hot swap policy and Auto Meta attributes can be changed by SMC.

Monitoring Configuration Sessions

• Config Session View

– All Active Tasks tab

- Will show any active configuration tasks that have been initiated by SMC
- Will not show configurations tasks executed via SYMCLI



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 5

Before initiating any configuration tasks from SMC it would be prudent to check if other configuration tasks have been initiated by other SMC users. The “All Active Tasks” tab of the Config Session view of a particular Symmetrix will show you if any tasks are currently active or running. Please note that this view only shows tasks initiated by this SMC server. Configurations initiated by SYMCLI will not be seen in this view.

Note: The use of the Config Session view is covered extensively in the pre-requisite Symmetrix Management Fundamental elearning course and will not be discussed any further in this class.

Set Symmetrix Attributes

- Symmetrix Admin → Set Symmetrix Attributes

Pre V-Max

Symmetrix Admin - Set Symmetrix Attributes (000190301138)

Maximum Number of Hypers per Disk : 256

☐ Auto Meta

Minimum Auto Meta Size : 65521 Cyl

Auto Meta Member Size : 0 Cyl (0 MB / 0.00 GB)

Auto Meta Configuration :

☒ Dynamic RDF Configuration

☒ Concurrent RDF Configuration

☒ Concurrent Dynamic RDF Configuration

SRDF/A Maximum Cache Usage (Percent) : 94

SRDF/A Maximum Host Throttle (Secs) : 0

Hot Swap Policy : Permanent

Add to Config Session List Cancel Help

V-Max

Symmetrix Admin - Set Symmetrix Attributes (000194900180)

Maximum Number of Hypers per Disk : 512

☐ Auto Meta

Minimum Auto Meta Size : 262669 Cyl

Auto Meta Member Size : 0 Cyl (0 MB / 0.00 GB)

Auto Meta Configuration :

SRDF/A Maximum Cache Usage (Percent) : 80

SRDF/A Maximum Host Throttle (Secs) : 0

Hot Swap Policy : Permanent

Add to Config Session List Cancel Help

Maximum number of hypers per disk
cannot be changed for V-Max

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 6

To set Symmetrix Attributes:

Right-click on a Symmetrix, choose “Symmetrix Admin” and then “Set Symmetrix Attributes ...” to launch the dialog shown. Make the desired changes and click the “Add to Config Session List”. This will list the task in the Config Session view from where the action can be committed.

For V-Max arrays the only attributes that can be changed are:

SRDF/A Maximum Cache Usage

SRDF/A Host Throttle

Hot Swap Policy

Auto Meta attributes

For pre V-Max arrays Dynamic, Concurrent and Concurrent Dynamic RFD configuration could be enabled or disabled.

Front End Port Attributes – Current Settings

Properties - 000194900180 Local/FA-7E/0

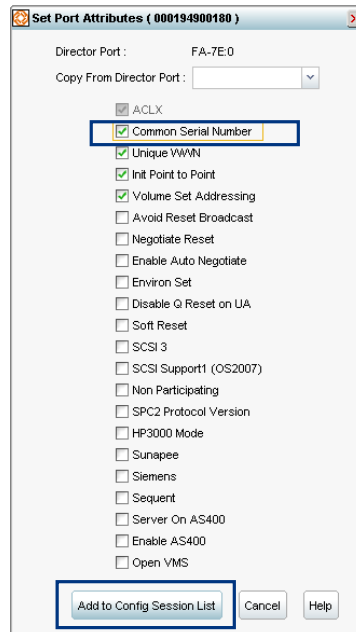
General		Fibre HBAs (4)	Mapped Devices (65)	Port Groups (1)	Masking Views (1)
Port Status	ON				
Number of Volumes	65				
WWN	50000972c002d118				
Fibre Channel Loop ID	N/A				
VCM State	N/A				
ACLX	Enabled				
Common Serial Number	Disabled				
Unique WWN	Enabled				
Init Point to Point	Enabled				
Volume Set Addressing	Enabled				
Avoid Reset Broadcast	Disabled				
Negotiate Reset	Disabled				
Auto Negotiate	Disabled				
Environ Set	Disabled				
Disable Q Reset on UA	Disabled				
Soft Reset	Disabled				
SCSI 3	Disabled				
SCSI Support1 (OS2007)	Disabled				
Non Participating	Disabled				
SPC2 Protocol Version	Disabled				

Can be set by SMC

The properties view of a front end port will show the attributes that are currently set. Many of these port attributes can be changed by SMC. In this example we see that the Common Serial Number bit is disabled for FA 7E:0.

Set Port Attributes

• Port and Director Configuration → Set Port Attributes



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 8

To set Port Attributes:

Right-click on a Symmetrix Port, choose “Port and Director Configuration” and then “Set Port Attributes ...” to launch the dialog shown. Make the desired changes and click the “Add to Config Session List”. This will list the task in the Config Session view from where the action can be committed.

In this example we are enabling the ‘Common Serial Number’ bit.

Lesson 2 – Configure Symmetrix Devices

Upon completion of this lesson, you will be able to use SMC to:

- Create and delete Symmetrix devices
 - Regular, Thin, Data and Save devices
- Form and dissolve Symmetrix meta volumes
- Set device attributes



The creation of Regular devices and FBA Meta devices with SMC was covered in the Symmetrix Management Console Fundamentals eLearning.

Create Symmetrix Device - Regular

Device Configuration → Create Device...

Configuration Options:

- 2-Way Mir
- 4-Way Mir
- BCV
- 2-Way BCV Mir
- BCV+R-5
- BCV+R-6
- RAID-5
- RAID-6
- RDF1
- RDF2
- RDF1+Mir
- RDF2+Mir
- RDF1+R-5
- RDF2+R-5
- RDF1+R-6
- RDF2+R-6
- DRV
- VDEV
- Unprotected

Emulation Options:

- FBA
- CELERRA_FBA
- VME_512_FBA
- AS400_M2107_A06
- AS400_M2107_A86
- AS400_M2107_A07
- AS400_M2107_A87
- AS400_M4328_50
- AS400_M2105_A01
- AS400_M2105_A81
- AS400_M2105_A02
- AS400_M2105_A82
- AS400_M2105_A04
- AS400_M2105_A84
- AS400_M2105_A05
- AS400_M2105_A85
- AS400_M6718_50
- AS400_M4326_50
- AS400_M4327_50
- AS400_M6717_50

Buttons: Add to Config Session List, Cancel, Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 10

Let us take a look at creating Symmetrix Devices. Right-click on a Symmetrix, choose “Device Configuration” and then “Create Device...” to launch the dialog shown. The Create Device dialog is used to create Regular, Thin, Data and Save devices. Thin and Data devices are used for Virtual Provisioning. Save devices are used by TimeFinder/Snap and by SRDF/A Delta Set Extension.

To create a Regular device click on the Regular Device tab and enter the number of devices, the Configuration (2-way Mir, RAID-5, RAID-6, BCV, etc.), the Data Member Count: This is only relevant for RAID-5 (3+1 or 7+1) or RAID-6 (6+2 or 14+2), the Emulation (FBA, CELERRA_FBA, etc.), Device capacity in cylinders – The drop down list will show you the existing devices sizes on the Symmetrix. Disk Group Number: Choose the disk group number if multiple disk groups have been configured. Dynamic Capability: The Dynamic RDF capability will allow the device to be made into a Dynamic SRDF device later if needed.

Additional input may be required if the Auto Meta feature is turned on or for RDF device configuration.

Click on “Add to Config Session List” to add this configuration request to the Config Session view from where the action can be committed.

Create Symmetrix Device - Thin

Device Configuration → Create Device...

The screenshot shows the 'Device Configuration - Create Device (000194900180)' dialog box. The 'Thin Device' tab is selected. The configuration is as follows:

- Number of Devices: 10
- Configuration: TDEV
- Emulation: FBA
- Capacity for each device: 1150 Cyl (1078 MB / 1.05 GB)
- Auto Meta Member Size: 0 Cyl (0 MB / 0.00 GB)
- Auto Meta Configuration: (blank)
- Dynamic Capability: DynRdf
- SCSI3 Persistent Reservation: Disabled
- RDF Group Number: (blank)
- Remote Configuration: TDEV
- Remote SSD: (blank)
- Bind to Pool: (blank)
- Remote Pool: (blank)
- Preallocated Capacity for each device: MB

Annotations point to the following elements:

- TDEV**: Points to the 'Configuration' dropdown, with a list showing 'RDF1+TDEV' and 'RDF2+TDEV'.
- FBA**: Points to the 'Emulation' dropdown, with a list showing 'CELERRA_FBA' and 'VME_512_FBA'.
- Option to bind Thin device to a Thin pool at creation**: Points to the 'Bind to Pool' field.
- Option to specify preallocated capacity**: Points to the 'Preallocated Capacity for each device' field.
- Add to Config Session List**: Points to the button at the bottom of the dialog.

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 11

To create a Thin device click on the Thin Device tab in the Device Configuration – Create Device dialog and enter the number of devices, the Configuration (TDEV, RDF1+TDEV or RDF2+TDEV), the Emulation (FBA, CELERRA_FBA, etc.), Device capacity in cylinders – The drop down list will show you the existing devices sizes on the Symmetrix. The device creation dialog also has an option to bind the new thin devices to a Thin Pool and to specify the preallocated capacity. In this example a Thin pool does not exist hence the Bind to pool entry is left blank.

The other options are similar to that of Regular devices seen on the previous slide.

Click on “Add to Config Session List” to add this configuration request to the Config Session view.

Create Symmetrix Device - Data

Device Configuration → Create Device...

The screenshot shows the 'Device Configuration - Create Device (000194900180)' dialog box. It has four tabs: 'Regular Device', 'Thin Device', 'Data Device' (selected), and 'Save Device'. The 'Data Device' tab contains the following fields:

- Number of Devices: 16
- Configuration: RAID-6 (dropdown menu)
- Data Member Count: 6 + 2 (dropdown menu)
- Emulation: FBA (dropdown menu)
- Capacity for each device: 1150 (dropdown menu) Cyl (1078 MB / 1.05 GB)
- Disk Group Number: Any Available Disk (dropdown menu)
- Device State in Pool: Disabled (dropdown menu)
- Add to Pool: (empty text box) Select ... (button)

Annotations with callouts point to the following elements:

- A box containing '2-Way Mir', 'RAID-5', and 'RAID-6' points to the 'Configuration' dropdown.
- A box containing 'Disabled' and 'Enabled' points to the 'Device State in Pool' dropdown.
- A box containing the text 'Option to add Data device to a Thin pool at creation' points to the 'Add to Pool' text box and the 'Select ...' button.
- The 'Add to Config Session List' button at the bottom is circled.

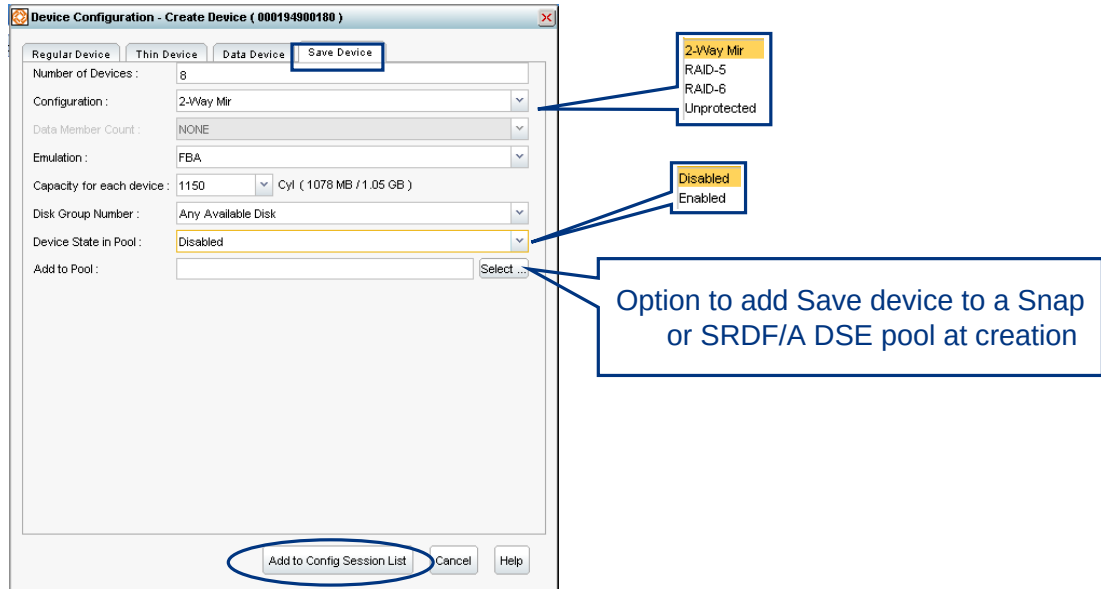
© 2009 EMC Corporation. All rights reserved. Configuration with Symmetrix Management Console - 12

To create a Data device click on the Data Device tab in the “Device Configuration – Create Device” dialog and enter the number of devices, the Configuration (2-Way Mir, RAID-5 or RAID-6), the Emulation (FBA only), Device capacity in cylinders – The drop down list will show you the existing devices sizes on the Symmetrix. The device creation dialog also has an option to add the new Data devices to a Thin pool. In this example a the entry is left blank. The device can also be Enabled (Disabled is the default) when added to the Thin pool.

Click on “Add to Config Session List” to add this configuration request to the Config Session view.

Create Symmetrix Device - Save

Device Configuration → Create Device...



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 13

To create a Save device click on the Save Device tab in the “Device Configuration – Create Device” dialog and enter the number of devices, the Configuration (2-Way Mir, RAID-5, RAID-6 or unprotected), the Emulation (FBA, CELERRA_FBA, etc.), Device capacity in cylinders – The drop down list will show you the existing devices sizes on the Symmetrix. The device creation dialog also has an option to add the new Save devices to a Snap or SRDF/A DSE pool. In this example the entry is left blank. The device can also be Enabled (Disabled is the default) when added to the Snap or DSE pool.

Click on “Add to Config Session List” to add this configuration request to the Config Session view.

Device Creation - Commit

ConfigSession - 000194900180 Local

My Active Tasks | My Inactive Tasks | All Active Tasks | All Inactive Tasks

ID	Config Item	Managed Object	Status	Description
1	Create Devices	000194900180	CREATED	Create 10 Dynamic RDF Capable devices of (RAID-6 6 + 2)(FBA)(capacity for e
2	Create Devices	000194900180	CREATED	Create 10 Dynamic RDF Capable thin devices of (FBA)(capacity for each device
3	Create Devices	000194900180	CREATED	Create 16 DATA devices of (RAID-6 6 + 2)(FBA)(capacity for each device = 11

Deactivate Preview All **Commit All** Abort All Activate Clear

Question
? Committing a config change may take a long time, continue?
Yes No

Wed Jun 10 11:34:47 EDT 2009 SymConfigChange session started

Wed Jun 10 11:38:48 EDT 2009 New Device(s) : 0191-019A on Disk Group Number 1, 019B-01A4, 01A5 on Disk Group Number 2, 01A6-01A9 on Disk

Wed Jun 10 11:38:48 EDT 2009 End Symm Config Session

Wed Jun 10 11:38:48 EDT 2009 SymConfigChange session ended Symmetrix=000194900180

Start configuration session

Configuration complete

© 2009 EMC Corporation. All rights reserved. Configuration with Symmetrix Management Console - 14

In this example three device “Create Devices” sessions have been added to the task list. 10 Regular Raid-6 devices, 10 Thin devices and 16 Data (Raid-6) devices. Click on Commit All to start the device creation.

The bottom part of the slide excerpts of the Config Session log. In this example we can see that the creation of these devices took about four minutes.



Creating an FBA Meta Device

FBA Meta Device Configuration → Form Meta...

FBA Meta Device Configuration - Form FBA Meta (000194900180)

Device Type: Regular
 Meta Config: Concatenated
 Stripe Size (cyl / blocks) : 2 / 1920
☐ Auto Select Number of Metas: 1 Number of Meta Members including Head: 1

Available Unmapped Devices: FILTERED 20 of 20 items

Dev	Cap (MB)	Dev Config	Emulation
0199	1078	RAID-6 (6 + 2)	FBA
019A	1078	RAID-6 (6 + 2)	FBA
019B	1078	TDEV	FBA
019C	1078	TDEV	FBA

Page 1 of 1 Items 1 to 20 of 20

Add devices to form meta

☐ Reserve Comments: Add to Config Session List

FBA Meta Device Configuration - Form FBA Meta (000194900180)

Device Type: Regular
 Meta Config: Concatenated
 Stripe Size (cyl / blocks) : 2 / 1920
☐ Auto Select Number of Metas: 1 Number of Meta Members including Head: 1

Available Unmapped Devices: FILTERED 6 of 6 items

Dev	Cap (MB)	Dev Config	Emulation
019F	1078	TDEV	FBA
01A0	1078	TDEV	FBA
01A1	1078	TDEV	FBA
01A2	1078	TDEV	FBA
01A3	1078	TDEV	FBA

Page 1 of 1 Items 1 to 6 of 6

Meta Members

Dev	Cap (MB)	Dev Config
019B	1078	TDEV
019C	1078	TDEV
019D	1078	TDEV
019E	1078	TDEV

Page 1 of 1 Items 1 to 4 of 4

Meta Head: 019B

☐ Reserve Comments: Expiration: Never

Add to Config Session List Cancel Help

© 2009 EMC Corporation. All rights reserved. Configuration with Symmetrix Management Console - 15

To create FBA Meta Devices. Right-click on a Symmetrix and choose “FBA Meta Device Configuration,” then choose “Form Meta...”.

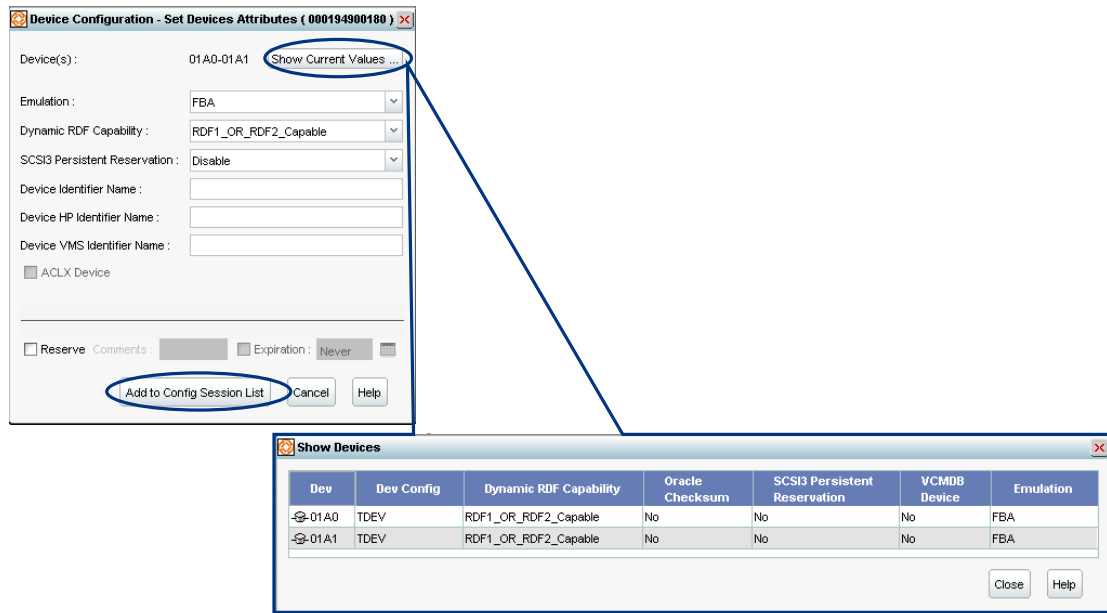
This dialog can be used to create Meta devices with regular (thick) or thin devices. Thin and thick devices cannot be members of the same meta device. The Meta Configuration can be Concatenated or Striped. In this example we are creating a Concatenated Thin meta device.

Highlight the devices that should form the meta from those listed in the unmapped devices list and click “Add”. This moves the devices to the Meta members’ column. The meta head can then be specified. As with all configuration tasks, click on the Add to Config Session List button. The actual commit of this action is done from ConfigSession view.

When creating a meta, you can optionally use the “Auto Select” feature. This allows you to specify only the number of metas, number of meta members per meta, and the meta heads; the Symmetrix microcode automatically chooses the meta members from the available pool of unmapped devices. At the time of the writing of the course the “Auto Select” feature does not apply if thin meta devices are to be created.

Changing Device Attributes

Device Configuration → Set Device Attributes ...



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 16

The following devices attributes can be changed:

Emulation: Only allowed for FBA devices (change between supported FBA types)

Dynamic RDF Capability

SCSI 3 Persistent Reservation

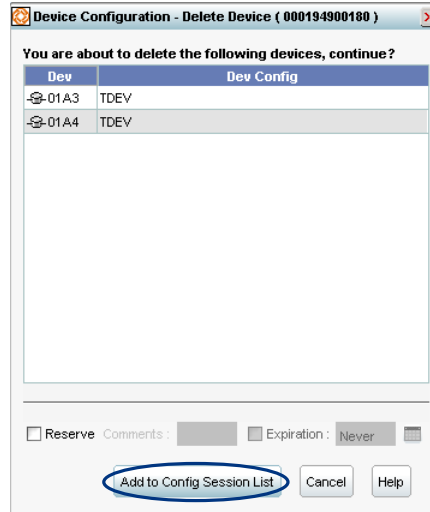
To change device attributes, highlight the desired device or devices and then choose Device Configuration → Set Device Attributes. This will launch the dialog shown on the screen.

Click on the “Show Current Values” button to see the attributes that are currently set on the devices.

Set the desired attribute and then click on “Add to Config Session List”. The task can be committed from the Config Session view.

Deleting Devices

Device Configuration → Delete Device



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 17

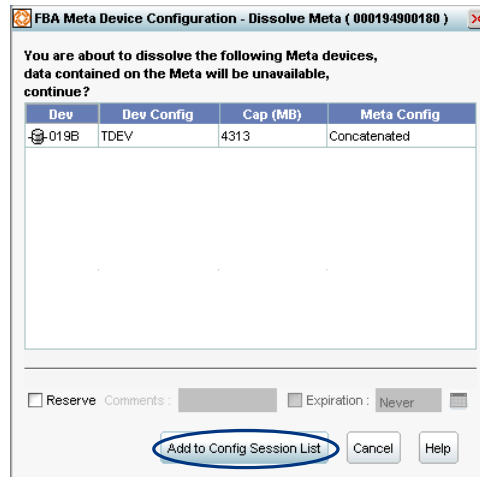
All the caveats discussed in the SYMCLI section for deleting devices apply to SMC as well.

To delete devices, highlight the desired device or devices and then choose Device Configuration → Delete Device. This will launch the dialog shown on the screen. SMC will automatically gray out the choice to delete a device if the device violates any of the rules for deletion.

Click on “Add to Config Session List”. The task can be committed from the Config Session view.

Dissolving Meta Devices

FBA Meta Device Configuration → Dissolve Meta...



All the caveats discussed in the SYMCLI section for dissolving meta devices apply to SMC as well.

To dissolve a meta device, highlight the desired meta device or devices and then choose FBA Meta Device Configuration → Dissolve Meta. This will launch the dialog shown on the screen.

Remember that all data on the meta device will be lost when the meta is dissolved.

Click on “Add to Config Session List”. The task can be committed from the Config Session view.

Demonstrations

- [01 Create Thin and Data devices](#)
- [02 Create Thin Meta device](#)

Click the hyperlinks to view the demos.

In Demo 01 SMC is used to create 10 Thin devices and 16 Data devices.

In Demo 02 A Thin Concatenated Meta device is created

Lesson 3 – Manage Device Pools

Upon completion of this lesson, you will be able to use SMC to:

- Create and manage device pools for TimeFinder/Snap, SRDF/A DSE and Virtual Provisioning
- Bind/unbind thin devices



Create Device Pool

Device Pool Management → Create Device Pool

Device Pool Management - Create Device Pool (000194900180)

Pool Name : VP_Pool1

Pool Type : Thin

Emulation : FBA

☒ Max Subscription % (0 -- 65534) : 150

☒ Enable New Pool Member

Available **FILTERED 8 of 8 items**

Pool	Dev	State	Cap (M)
N/A	01AD	Disabled	
N/A	01AE	Disabled	
N/A	01AF	Disabled	
N/A	01B0	Disabled	
N/A	01B1	Disabled	
N/A	01B2	Disabled	
N/A	01B3	Disabled	
N/A	01B4	Disabled	

Target

Pool	Dev	State	Cap (M)
N/A	01A5	Disabled	
N/A	01A6	Disabled	
N/A	01A7	Disabled	
N/A	01A8	Disabled	
N/A	01A9	Disabled	
N/A	01AA	Disabled	
N/A	01AB	Disabled	
N/A	01AC	Disabled	

Page 1 of 1 Items 1 to 8 of 8

Total Enabled Pool Capacity : 8625.00 MB

OK Cancel Help

© 2009 EMC Corporation. All rights reserved.

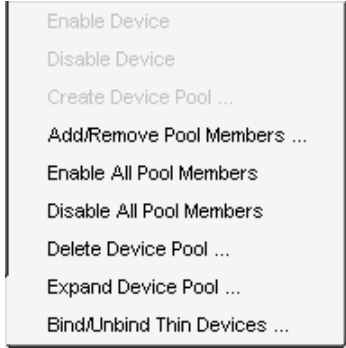
Configuration with Symmetrix Management Console - 21

Snap, SRDF/A DSE and Thin pools can be created by right clicking on a Symmetrix (or the Pools folder under a Symmetrix) and choosing Device Pool Management → Create Device Pool.

Enter a name, choose the pool type (Snap, SRDF/A DSE or Thin). For Thin pools the maximum subscription limit can set. Optionally one can add Save or Thin devices (base on the Pool Type) to the pool and enable them. To add devices to the pool, pick the devices from the Available column and click on the Add button to move them the Target column. The Available column will show Save or Data devices based on the Pool Type. Only disabled Save/Data devices will be shown.

Other Pool Management Activities

- Enable/Disable specific devices
- Enable/Disable all members
- Add/Remove members
- Expand device pool
- Bind/Unbind Thin Devices
- Delete device pool



Enable Device
Disable Device
Create Device Pool ...
Add/Remove Pool Members ...
Enable All Pool Members
Disable All Pool Members
Delete Device Pool ...
Expand Device Pool ...
Bind/Unbind Thin Devices ...

In addition to creating device pools, the tasks listed on the slide can also be performed. Devices have to be disabled before they can be removed from a pool.

Expand Device Pool

Device Pool Management → Expand Device Pool

Device Pool Management - Expand Device Pool (00019490018)

Pool Name : VP_Pool1

Expand Pool by :

☒ 100 %

☐ GB

Method :

☒ Create New Devices

☐ Use Non-Pooled Available Devices

Note: The capacity, config and disk group number of the first device in the pool are 1078 MB, RAID-6 (6 + 2) and 2. They will be used to create new devices.

Add to Config Session List Cancel Help

A Device Pool can be expanded by right clicking on the Pool and choosing Device Pool Management → Expand Device Pool.

Specify how much to expand the pool by (in percentage or GB) and the method. The method can be to either create new devices or to use existing Save/Data devices which do not belong to a pool. The configuration of the new devices will be identical to that of the first device in the pool. Creation of new devices will lead to a Config Session task.

Bind/Unbind Thin devices to Thin Pool

Device Pool Management → Bind/Unbind Thin Devices

Device Pool Management - Bind/Unbind Thin Devices (000194900180)

Pool Name : VP_Pool1 Pool Type : Thin Pool Emulation : FBA

Preallocated Capacity for each new device : 100 MB Total Enabled Pool Capacity : 8625.00 MB

Available FILTERED 6 of 6 items

Device	Cap (MB)	Dev Config	R
019F	1078	TDEV	
01A0	1078	TDEV	
01A1	1078	TDEV	
01A2	1078	TDEV	
01A3	1078	TDEV	
01A4	1078	TDEV	

Target

Device	Cap (MB)	Dev Config	R
019B	4313	TDEV	

Page 1 of 1 Items 1 to 6 of 6 Page 1 of 1 Items 1 to 1 of 1

Max Subscription % : 150 Current subscription % : 50.00 Total Bound Capacity : 4312.50 MB

☐ Reserve Comments : ☐ Expiration : Never

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 24

To Bind or Unbind Thin devices to a pool, right click on the thin pool and choose Device Pool Management → Bind/Unbind Thin Devices. All unbound thin devices will be listed in the Available column. To bind a device highlight it in the Available column and click on the Add button to add it to the Target column. The preallocated capacity for the device can also be specified (in Cylinders, MB or GB). To unbind a device highlight it in the Target column and click on the Remove button. Like all the other configuration tasks click on the “Add to Config Session List” button and then go to the Config Session view to commit the task.

Demonstration

- [03 Create Thin Device Pool and Bind Thin Device](#)

Click on the link to view the demo.

In this demonstration a Thin Device Pool is created with 4 data devices and then a thin meta device is bound to this pool.

Lesson 4 – Autoprovisioning Groups

Upon completion of this lesson, you will be able to use SMC to:

- Create/Modify
 - Initiator groups
 - Port groups
 - Storage groups
 - Masking views
- Backup device masking information
- Set HBA flags
- Map/unmap devices



Autoprovisioning with SMC was covered in the Symmetrix Management Console Fundamentals impact, we are reiterating the concepts one more time in this lesson.

Initiator Group Creation

Device Masking and Mapping → Initiator Groups Maintenance

Create Initiator Group ...

Modify Initiator Group ...

Rename Initiator Group ...

Delete Initiator Group ...

Initiator Group Management - Create (000194900181)

Initiator Group Name : WIN2_Initiators

Available Initiators	
Initiator	Type
50060b00000783ba	VVWN
50060b0000084d28	VVWN
10000000c94eac7e	VVWN
10000000c94eac7f	VVWN
10000000c9210467	VVWN
10000000c92271fb	VVWN
10000000c941249f	VVWN
10000000c941873c	VVWN

Page 1 of 1 Items 1 to 8 of 8

New Initiator:

Set HBA Flags

Selected Initiators	
Initiator	Type
10000000c9422f93	VVWN
10000000c9422fc7	VVWN

Page 1 of 1 Items 1 to 2 of 2

OK Cancel Help

© 2009 EMC Corporation. All rights reserved.

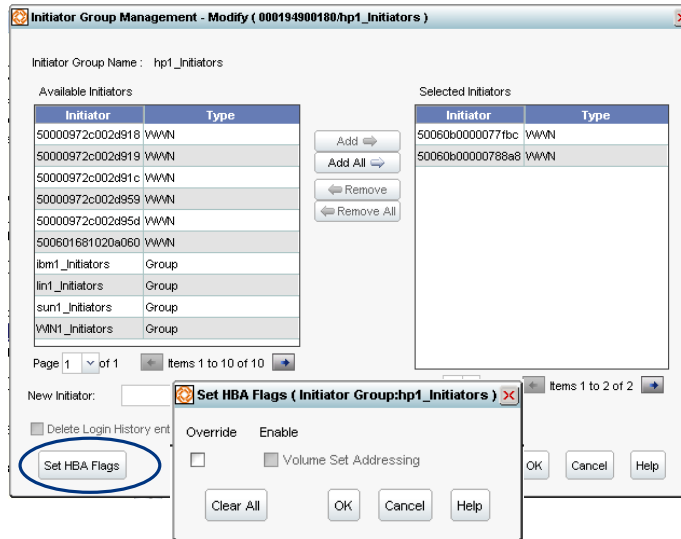
Configuration with Symmetrix Management Console - 27

To create a new Initiator Group right click on a Symmetrix V-Max choose Device Masking and Mapping choose Initiator Groups Maintenance and then Create Initiator Group. Give the initiator group a unique name, then choose the initiator or initiator groups from the Available Initiator column and add the same to the Selected Initiators columns and click Ok to create the initiator group. In this example the Initiator group WIN2_Initiators has two members.

Initiator Groups can be modified by choosing Device Masking and Mapping → Initiator Groups Maintenance → Modify Initiator Group

Modify Initiator Group

Device Masking and Mapping → Initiator Groups Maintenance → Modify Initiator Group



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 28

To modify an Initiator Group right click on the Initiator group and choose Device Masking and Mapping → Initiator Groups Maintenance → Modify Initiator Group.

Initiators or initiator groups can be added or removed as necessary. One can use the “Set HBA Flags” button to override some Port flag settings. Some of the other port flags can also be overridden on an initiator by initiator basis. We will see this at the end of this lesson.

Port Group Creation

Device Masking and Mapping → Port Groups Maintenance

Port Group Management - Create (000194900181)

Port Group Name : WIN2_StoragePorts

Available Ports

Port	# Port Groups	# Mappings
FA-7E:0	0	57
FA-7E:1	0	57
FA-7F:0	0	57
FA-8E:0	0	57
FA-8E:1	0	57
FA-8F:0	0	57

Page 1 of 1 Items 1 to 6 of 6

Group Members

Port	# Port Groups	# Mappings
FA-7F:1	0	113
FA-8F:1	0	113

Page 1 of 1 Items 1 to 2 of 2

OK Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 29

To create a new Port Group right click on a Symmetrix V-Max choose Device Masking and Mapping choose Port Groups Maintenance and then Create Port Group. Give the port group a unique name, then choose the Ports Available Ports column and add the same to the Group Members columns and click Ok to create the Port group. In this example the port group WIN2_StoragePorts has two members.

Port Groups can be modified by choosing Device Masking and Mapping → Port Groups Maintenance → Modify Port Group

Storage Group Creation

Device Masking and Mapping → Storage Groups Maintenance

Create Storage Group ...

- Modify Storage Group ...
- Expand Storage Group ...
- Rename Storage Group ...
- Delete Storage Group ...

Storage Group Management - Create (000194900181)

Storage Group Name : WIN2_StorageGroup

Devices Source Type : Symmetrix

Device Source Name : 000194900181

SSID : Select ...

Available Devices **FILTERED 56 of 281 items**

Dev	Cap (MB)	Dev Config	Emulation
013A	1078	2-Way Mir	
013B	1078	2-Way Mir	
013C	1078	2-Way Mir	
013D	1078	2-Way Mir	
013E	1078	2-Way Mir	
013F	1078	2-Way Mir	
0140	1078	2-Way Mir	

Group Members

Dev	Cap (MB)	Dev Config
-----	----------	------------

Page 1 of 1 Items 1 to 56 of 56

OK Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 30

To create a new Storage Group right click on a Symmetrix V-Max choose Device Masking and Mapping choose Storage Groups Maintenance and then Create Storage Group. Give the Storage group a unique name. Choose the Device Source Type: This can be Symmetrix, Device Group, Composite Group or a Storage Template. We will discuss Storage Templates shortly. In this example we have chosen Symmetrix as the Device Source and then chosen Symmetrix ID 181. Pick the devices from the available devices column and add them to the Group Member column. Click Ok to create the Storage group. In this example the Storage group WIN2_StorageGroup will have a number of Symmetrix Devices from Symmetrix 181. One can use a previously defined Storage Template when creating or expanding a Storage Group.

Masking View Creation

Device Masking and Mapping → Masking Views Maintenance

Create Masking View ...
 Rename Masking View ...
 Delete Masking View ...

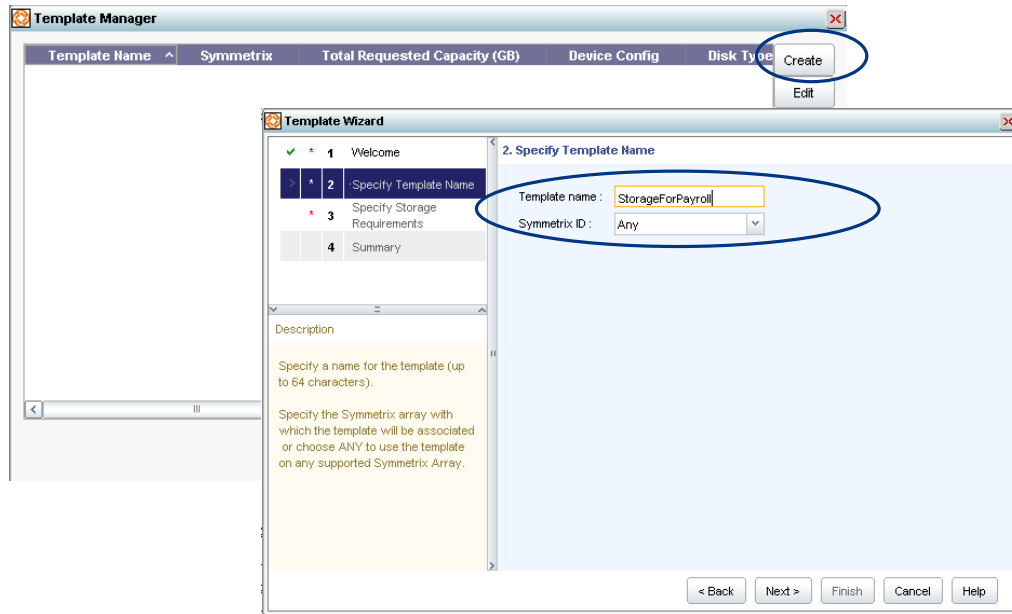
Un-mapped devices are automatically mapped to the appropriate front end ports

To create a new Masking View right click on a Symmetrix V-Max choose Device Masking and Mapping choose Masking Views Maintenance and then Create Masking View. Give the Masking View a unique name. Choose existing Storage, Port and Initiator groups by clicking on the appropriate Select button or create New groups by clicking on the appropriate New Button. In this example we have used the Select button to pick the Storage, Port and Initiator groups that we had created already. Click on the “Set Dynamic LUN Address” to manually change the host LUN addresses. By default the next available host LUN address will be automatically assigned to the devices.

Click OK to complete the creation of the Masking View. As soon as the Masking View is created the host reconfiguration steps can be performed and the host should see the V-Max devices.

Storage Templates

Tasks → Template Wizard



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 32

SMC allows the creation of Storage Templates which can be used later on when provisioning storage to V-Max arrays. Storage Templates are available in SMC only. To create a new template change the SMC view to the Tasks View and then click on the Template Wizard Hyperlink. Click on the Create button in the Template Manager window to launch the Template Wizard. The first step is the Welcome page (not shown on slide). In the second step the name of the Template and the Symmetrix ID is specified (one can choose "Any" for the Symmetrix ID).

Storage Templates ...

Template Wizard

1 Welcome
2 Specify Template Name
3 Specify Storage Requirements
4 Summary

3. Specify Storage Requirements

Template name : StorageForPayroll

Symmetrix ID : Any

Total requested capacity : 20 GB

* Devices : ☐ Find Existing ☒ Create ☐ Best Effort

* Device configuration : RAID-6 6 + 2

Emulation : FBA

* Capacity for each device : 1150 Cyl (1078 MB / 1.05 GB)

* Disk : Fibre

RDF : ☐ None ☒ Dynamic capability

Preallocate per device : %

Meta : ☐ Yes ☒ No

Meta member count (including head) :

Meta configuration : Striped

< Back Next > Finish Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 33

In the third step the Storage Requirements are specified.

Total requested capacity: Specify the total capacity required in Gigabytes.

Devices: The template can find existing devices in the array or create new devices or it can do a combination of both if “Best Effort” is chosen. In the example shown the “Create” option has been chosen.

Device Configuration: Choose the configuration from the drop down list – We have chosen RAID-6 6+2 in this example.

Emulation: FBA, CKD etc. In this case we have chosen FBA.

Capacity for each device: Specify this in cylinders.

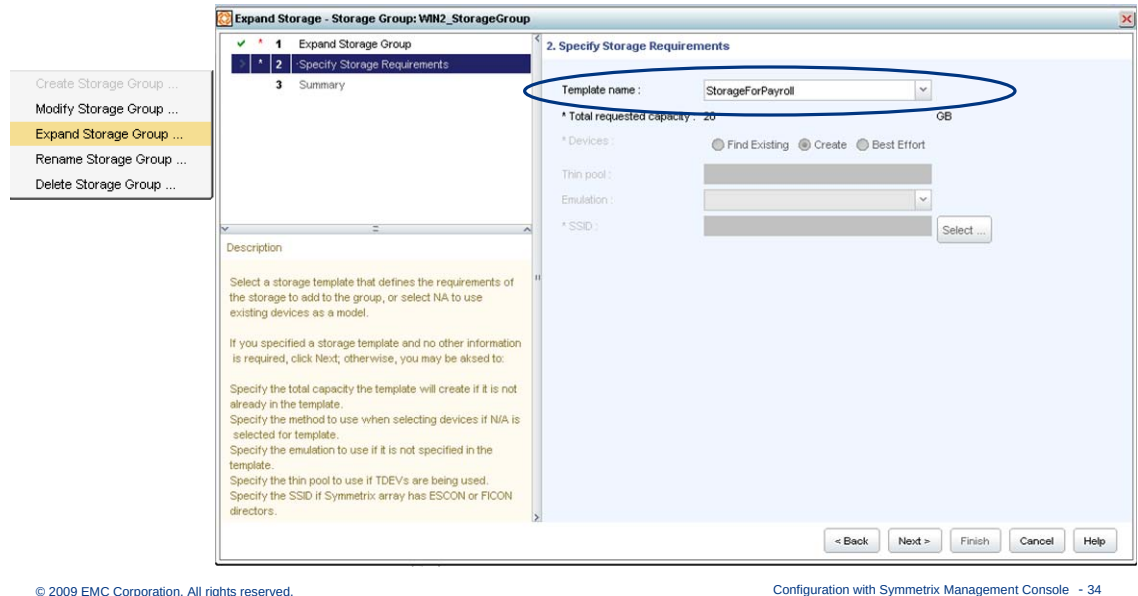
Disk: One can choose between the various disk types, Fibre, SATA or Flash (of course such disks must exist on the array).

In addition one can also specify the Dynamic RDF capability and Meta configuration information if Meta devices are desired. Pre-allocation percentage can also be specified for Thin Devices.

The next step will show a summary (not shown on slide). Click on the Finish button to complete the creation of the template.

Storage Templates ...

- Storage Template can be used
 - When creating a new Storage Group or expanding a Storage Group



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 34

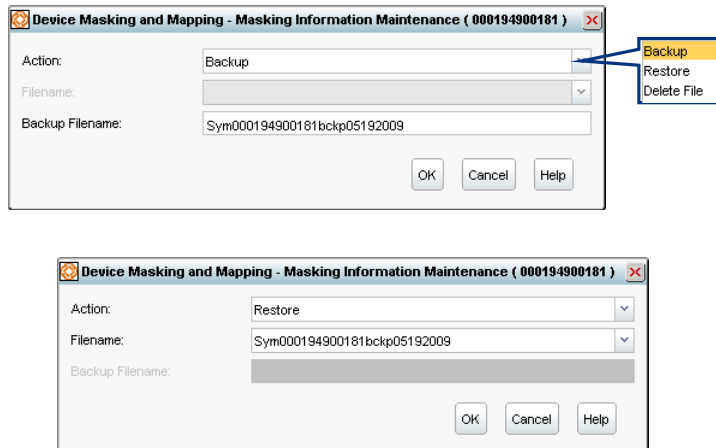
Storage Templates can be used when creating a new Storage Group or when expanding an existing Storage Group.

To expand an existing Storage Group, right click on the existing Storage Group and choose Device Masking and Mapping, then choose Storage Groups Maintenance and then Expand Storage Group. Select a storage template that defines the requirements of the storage to add to the group, or select NA to use existing devices as a model. In this example we have chosen the template "StorageForPayroll". If you specify a storage template then no other information is required, click Next for the summary and then Click on Finish.

If a template is not used (N/A) then specify the total capacity to create. Specify the method to use when selecting devices, the emulation to use and the thin pool to use if TDEVs are being used. One may also have to specify the SSID if Symmetrix array has ESCON or FICON directors.

Backup/Restore Autoprovisioning Groups

Device Masking and Mapping → Masking Information Maintenance...



Masking Information Maintenance Operations for V-Max arrays are performed by right-clicking a Symmetrix and choosing “Device Masking and Mapping...”, then “Masking Information Maintenance...”. This launches the Masking Information Maintenance dialog.

Choose “Backup” from the Action dropdown for a backup, and then specify a backup file name and click “OK.”

Choose “Restore” for a restore operation, then choose a filename from which the restore is to be performed,

Map/Unmap Devices

- Map Devices

- Device Masking and Mapping → Map Devices...
- On V-Max arrays devices are automatically mapped if necessary during the
 - creation of a Masking view
 - modification of a Storage group which belongs to a Masking view
 - modification of a Port group which belongs to a Masking View

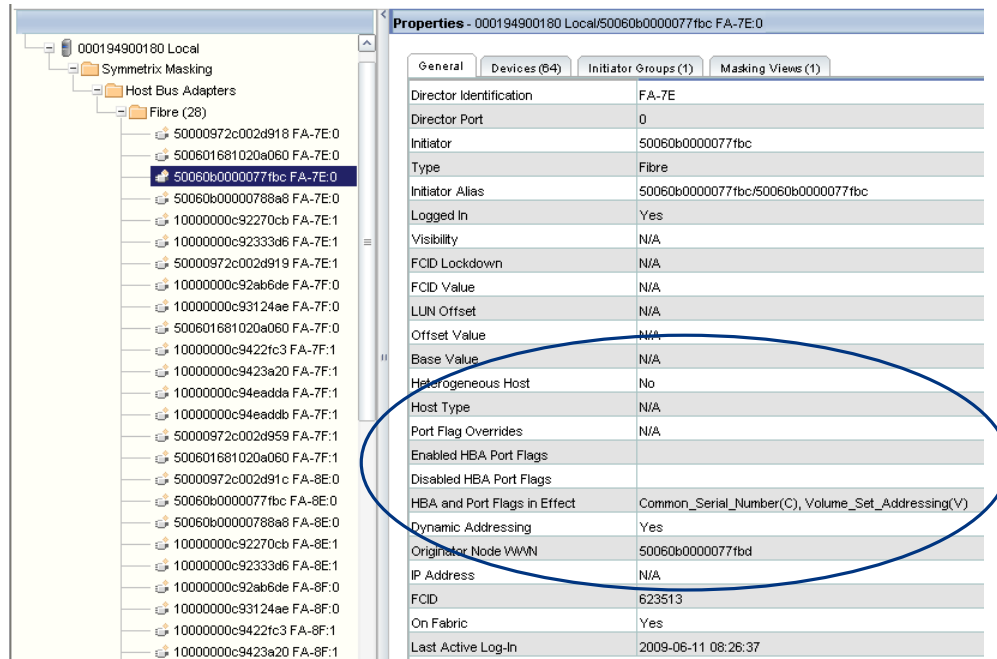
- Unmap Devices

- Device Masking and Mapping → Unmap Devices...
- On V-Max arrays devices can be optionally unmapped during the
 - deletion of a Masking view
 - modification of a Storage group which belongs to a Masking view
 - modification of a Port group which belongs to a Masking View

Mapping was covered in the Symmetrix Management Console Fundamentals, thus we will not cover it in detail here.

Explicit mapping and unmapping of devices is not necessary on V-Max arrays with Autoprovisioning Groups.

HBA Flags – Current Settings



Properties - 000194900180 Local/50060b0000077fbc FA-7E:0

General	
Director Identification	FA-7E
Director Port	0
Initiator	50060b0000077fbc
Type	Fibre
Initiator Alias	50060b0000077fbc/50060b0000077fbc
Logged In	Yes
Visibility	N/A
FCID Lockdown	N/A
FCID Value	N/A
LUN Offset	N/A
Offset Value	N/A
Base Value	N/A
Heterogeneous Host	No
Host Type	N/A
Port Flag Overrides	N/A
Enabled HBA Port Flags	
Disabled HBA Port Flags	
HBA and Port Flags in Effect	Common_Serial_Number(C), Volume_Set_Addressing(V)
Dynamic Addressing	Yes
Originator Node WWN	50060b0000077fbd
IP Address	N/A
FCID	623513
On Fabric	Yes
Last Active Log-In	2009-06-11 08:26:37

© 2009 EMC Corporation. All rights reserved.

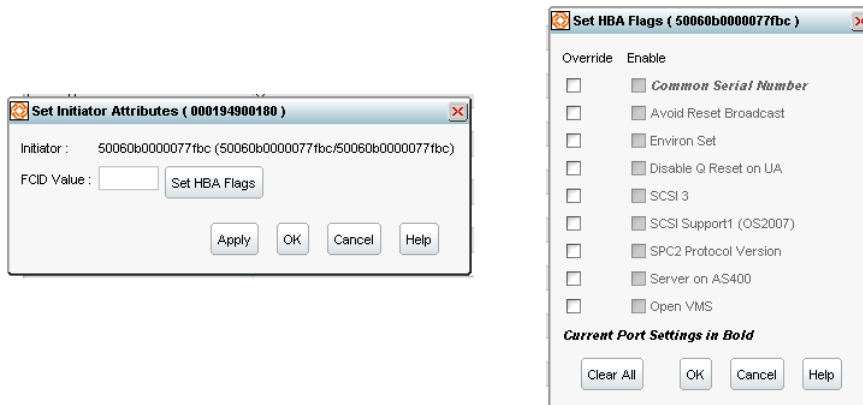
Configuration with Symmetrix Management Console - 37

HBA flags can be set on a initiator by initiator basis to override the port flag settings. The properties view of a particular initiator logged into a particular front end port will show the current HBA flags that are in effect.

In this example, this initiator which is logged into port FA-7E 0 is not using any port flag overrides and currently only the C and V bits are in effect.

HBA Flags – Set Initiator Attributes

Device Masking and Mapping → Set Initiator Attributes



To override port flag settings for an initiator, right click on a initiator (in the Fibre folder under Host Bus Adapters) and choose Device Masking and Mapping → Set Initiator Attributes. In the Set Initiator Attributes dialog box click on the Set HBA Flags button. The override the desired flags in the Set HBA Flags dialog and Click OK. Then click OK on the Set Initiator Attributes dialog box to finish the operation.

Lesson 5 – SRDF Configuration

Upon completion of this lesson, you will be able to use SMC to:

- Configure SRDF groups
- Create SRDF device pairs
- Set SRDF/A group attributes





SRDF Replication Wizard

Tasks View →  [SRDF Replication Wizard ...](#)

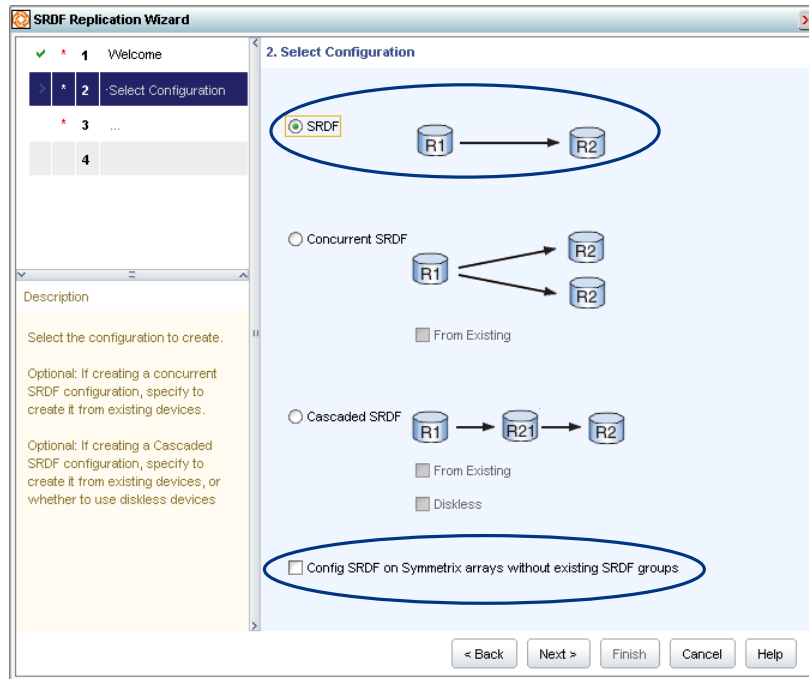


© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 40

The SMC SRDF Replication Wizard guides the user through the process of creating SRDF Groups and device pairs. The wizard is available on the Tasks view page. Click on the SRDF Replication Wizard hyperlink to start the wizard.

SRDF Replication Wizard – Select configuration



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 41

In this example we choose SRDF as the configuration. In this environment SRDF relationships already exist between the Symmetrix arrays managed by SMC, so we leave the ‘Config SRDF on Symmetrix arrays without existing SRDF groups’ unchecked. Click Next to proceed.

SRDF Replication Wizard – Source Devices

SRDF Replication Wizard

1 Welcome
2 Select Configuration
3 **Select Source Devices**
4 Select Symmetrix B
5 Verify Symmetrix B Devices
6 Summary

3. Select Source Devices

Symmetrix A : 000194900180
Group Type : Ungrouped
Group Name :

Ungrouped
Device Group
Composite Group
Storage Group

Available Devices **FILTERED 247 of 249 items**

Dev	Dev Config	Cap (MB)
0029	2-Way Mir	1T
002A	2-Way Mir	1T
002B	2-Way Mir	1T
002C	2-Way Mir	1T
002D	2-Way Mir	1T
002E	2-Way Mir	1T
002F	2-Way Mir	1T
0030	2-Way Mir	1T

Page 1 of 1 Items 1 to 247 of 247

Selected Source Devices

Dev	Dev Config	Cap (MB)
0191	RAID-6	
0192	RAID-6	

Page 1 of 1 Items 1 to 2 of 2

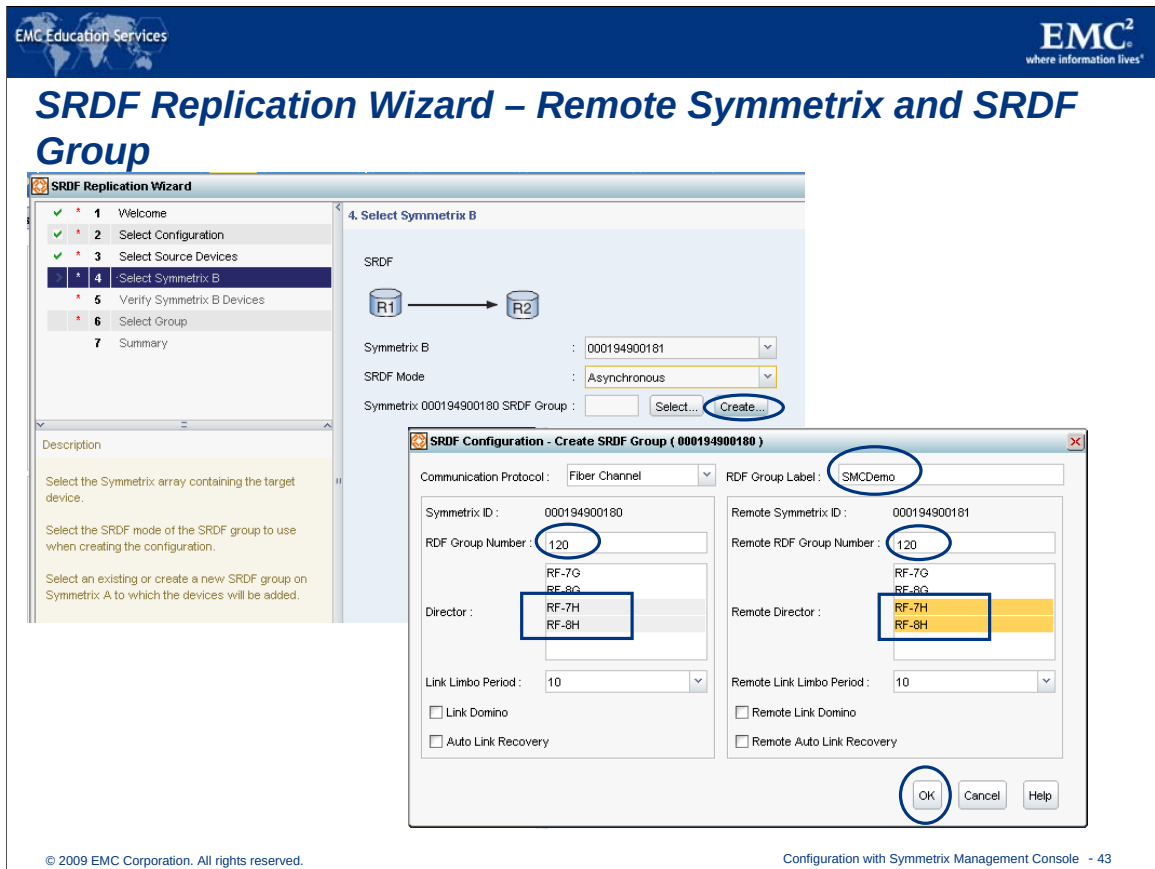
< Back Next > Finish Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 42

In this step the source Symmetrix and devices have to be selected. Ungrouped devices or devices that belong to Device, Composite or Storage Groups can be chosen. In this example Ungrouped has been chosen. Two Raid-6 devices have been chosen from the Available Device column and added to the Selected Source Devices column.

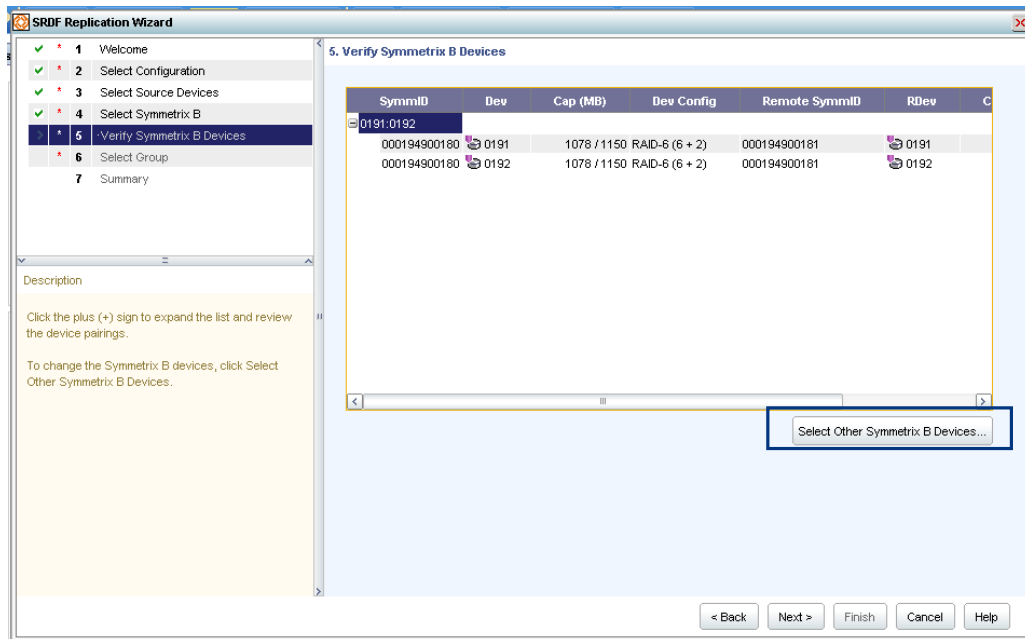
Click Next to proceed.



In this step the target Symmetrix and the SRDF mode and the SRDF group on the source Symmetrix have to be specified. A new SRDF group can be created by Clicking on the Create button. In the Create SRDF Group dialog, specify the RDF Group Label, RDF Group Number (Source and Remote) and the RDF Directors (Source and Remote) and Click OK. This new RDF group will be used by the Wizard.

SRDF Groups can also be created by right clicking on a Symmetrix and choosing Replication → SRDF Configuration → Create SRDF Group

SRDF Replication Wizard – Remote Devices



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 44

Based on the Source Devices the wizard automatically picks appropriate remote devices from the target Symmetrix. The selection can be changed by clicking on the “Select Other Symmetrix B Devices”. Click Next to continue.



SRDF Replication Wizard – Select Group

SRDF Replication Wizard

1 Welcome
2 Select Configuration
3 Select Source Devices
4 Select Symmetrix B
5 Verify Symmetrix B Devices
6 Select Group
7 Summary

6. Select Group

Group Type : Device Group

☒ Add Devices To New Group : SMCDemoDG

☐ Add Devices To Existing Group : Select...

Description

Select the type of group to which the SRDF pairs are to be added, if any.

If adding the pairs to a group, select whether to add them to a new or existing group.

If adding to a new group, specify a unique name (up to 31 characters) for the group.

If adding to an existing, specify the name of the group or click Select to select from a list of existing.

< Back Next > Finish Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 45

The ungrouped device pairs can be optionally added to a Device Group or a Composite group. In this example we will add it to a new device group. Click Next to continue.

SRDF Replication Wizard – Summary

SRDF Replication Wizard

7. Summary

Step Name

- 2 Select Configuration
 - SRDF
- 3 Select Source Devices
 - Symmetrix A : 000194900180
 - Group : Ungrouped
 - Group Name : N/A

Dev	Symm ID	Dev Config	Cap (MB)	Reserved
0191	000194900180	RAID-6 (6 + 2)	1078	No
0192	000194900180	RAID-6 (6 + 2)	1078	No
- 4 Select Symmetrix B
 - Symmetrix B : 000194900181
 - SRDF Mode : Asynchronous
 - SRDF Group : 120
- 5 Verify Symmetrix B Devices

Description

Review your previous selections.

To make changes, click Back to return the previous step.

Click Finish to create the configuration.

< Back Next > Finish Cancel Help

Review the Summary and click Finish to complete the SRDF Group and Pair Creation process.

SRDF Group, Pairs and Device Group Created

The screenshot displays the Symmetrix Management Console interface. On the left is a tree view of the storage configuration. The main area shows the 'Properties' window for the SRDF group '000194900180 Local/RF-7H/120'. Below it, the 'Properties' window for the device group 'SMCDemoDG [900180]' is shown, specifically the 'RDF Query (2)' tab.

Properties - 000194900180 Local/RF-7H/120

General	
RDF (RA) Group Number	120
RDF (RA) Group Label	SMCDemo
Remote RDF (RA) Group Number(s)	120
Remote Symmetrix ID(s)	000194900181
Remote RDF (RA) Director Number(s)	120, 119
Remote Director Ident(s)	RF-8H, RF-7H
RDF (RA) Group Flags	Dynamic
Prevent Auto Link Recovery	Enabled
Prevent RAs Online Upon Power On	Enabled
Link Domino	Disabled
Link Config	Fibre
Director Config	Switch
RDF (RA) Group Configuration	Dynamic
Farpoint	No
RDF/A Flags	SessionActive, Async, ConsistencyExemptDevices, R2NotConsistent
Link Limbo (sec)	10
Minimum Cycle Time	30
Session Priority	33
Transmit Idle Time	000:00:00
Transmit Idle	Enabled
R1 Side Percent Cache in Use	12
R2 Side Percent Cache in Use	0
Dynamic Cache Partition Name	DEFAULT_PARTITION
SRDF/A DSE Status	Inactive
SRDF/A DSE Autostart	Disabled

Properties - SMCDemoDG [900180]

RDF Query (2)								
LDev	Dev	Link Status	Remote SymmID	RDev	Pair State	R1 Inv	R2 Inv	MIDACE
DEV001	0191	Ready	000194900181	0191	SyncInProg	0	10035 A...X	
DEV002	0192	Ready	000194900181	0192	SyncInProg	0	8885 A...X	

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 47

The SRDF group, RDF device pairs and the device group have been successfully created. The SRDF/A group parameters can be seen in the properties view of the SRDF group. The RDF query tab of the properties view of the device group shows the pair state and the SRDF mode.

Setting SRDF/A Group Attributes

Replication → SRDF Configuration → Set SRDF/A Group Attributes

SRDF Configuration - Set SRDF/A Group Attributes (000194900180)

RDF Group : 120

RDFA Min Cycle Time (sec) : 30

RDFA Session Priority ("1" is the highest) : 33

Transmit Idle Time : ☒ enabled

SRDF/A DSE

SRDF/A DSE Pool Name (FBA) :

SRDF/A DSE Pool Name (CKD3390) :

SRDF/A DSE Pool Name (CKD3380) :

SRDF/A DSE Pool Name (AS400) :

Threshold : 50

Autostart : ☐ enabled

Activate/Deactivate SRDF/A DSE : No Change

OK Cancel Help

To set SRDF/A group attributes, right click on an SRDF Group and choose Replication → SRDF Configuration → Set SRDF/A Group Attributes. Make the desired changes and click on OK.

Demonstration

- [04 Create SRDF Group and Create SRDF Pairs](#)

Click on the link to view demo.

In this demonstration an SRDF Group is created. Then two SRDF Device pairs are created. The R1 devices are then added to an existing Storage Group. The lecture showed the Replication Wizard. In the demo we show an alternate method.

Lesson 6 – LUN Migration

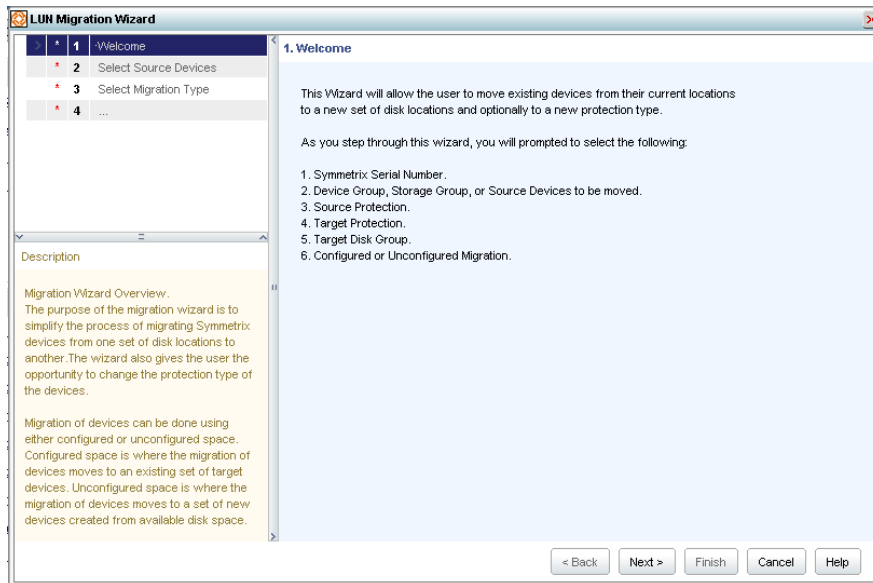
Upon completion of this lesson, you will be able to use SMC to:

- Perform LUN Migration
 - Using configured space
 - Using un-configured space



LUN Migration Wizard

Tasks →  [LUN Migration Wizard ...](#)



The SMC LUN Migration Wizard guides the user through the process of migrating Symmetrix devices from one set of disk locations to another. The wizard also gives the user the opportunity to change the protection type of the devices. Migration of devices can be done using either configured or unconfigured space. Configured space is where the migration of devices moves to an existing set of target devices. Unconfigured space is where the migration of devices moves to a set of new devices created from available disk space.

LUN Migration Wizard – Configured Space Example

• Current Configuration

Properties - SMCdemoDG [900180]/Standard Devices (2)

Dev	LDev	Dev Config	Cap (MB/Cyl)	Dev Status
0191	DEV001	RDF1+R-6 (6 + 2)	1078 / 1150	Ready
0192	DEV002	RDF1+R-6 (6 + 2)	1078 / 1150	Ready

Migrate two RDF1 RAID-6 (6+2) devices to RDF1 RAID1 (2-Way-Mir) using configured space

Properties - 000194900180 Local/Unmapped (10)

Dev	Dev Config	Cap (MB/Cyl)	Dev Status	Reserved	Emulation	# Paths
01B5	2-Way Mir	1078 / 1150	Ready	No	FBA	0
01B6	2-Way Mir	1078 / 1150	Ready	No	FBA	0

In this example we our production applications are running on devices 191 and 192 which are currently RAID-6. We want to convert these devices to RAID1 devices using the configured space of devices 1B5 and 1B6. We will use the LUN Migration Wizard.

LUN Migration Wizard – Select Source Devices

LUN Migration Wizard

1 Welcome

2 **Select Source Devices**

3 Select Migration Type

4 ...

5 Summary

Description

Refine the list of source devices by one or a combination of 3 ways. The devices can be selected by group type which consists of ungrouped, device group, storage group. If ungrouped is selected, the table will be populated with all the devices on the chosen symmetrix. If Device Group is selected, the option to select all the STDs, BCVs or TGTs or the group become available via the Select button. Once the selection has been made, the table will be populated with the devices which correspond to the selection criteria

Symmetrix ID : 000194900180

Session Name : MigrateToConfigured

Group Type : Device Group

Group Name : SMCDemoDG Select...

Ungrouped
Device Group
Storage Group

Dev	Dev Config	Cap (M)
0191	RDF1+R-6 6+2	1078 / 1150
0192	RDF1+R-6 6+2	1078 / 1150

Selected Source Devices

Dev	Dev Config	Cap (M)
-----	------------	---------

Page 1 of 1 Items 1 to 2 of 2

< Back Next > Finish Cancel Help

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 53

In this step the source devices to be migrated are specified. Choose the Symmetrix ID, give the migration session a name and then the devices to be migrated. Ungrouped devices or devices that belong to Device or Storage Groups can be chosen. In this example we have chosen Device Group and then chosen the SMCDemoDG device group (click the Select button to make the selection). Selecting the device group automatically populates the Selected Source Devices column. In this case devices 191 and 192 were members of the device group SMCDemoDG.

Click Next to proceed.

LUN Migration Wizard – Select Migration Type

LUN Migration Wizard

1 Welcome
2 Select Source Devices
3 **Select Migration Type**
4 ...
5 Summary

3. Select Migration Type

☒ Configured devices
 Protection Type : RAID1
 Disk Group : 2 (RPM 15000)

☐ UnConfigured devices
 Protection Type :
 Disk Group :

Description

Define the type of migration that is to be used and the protection of the devices to be migrated to.

If the Configured Space radio button is selected, the corresponding options for configured space will become available. These options allow the Protection Type and the source of the potential Target Devices to be specified.

If the Unconfigured Space radio button is selected, then Protection Type and Target Disk Group need to be specified.

< Back Next > Finish Cancel Help

© 2009 EMC Corporation. All rights reserved.

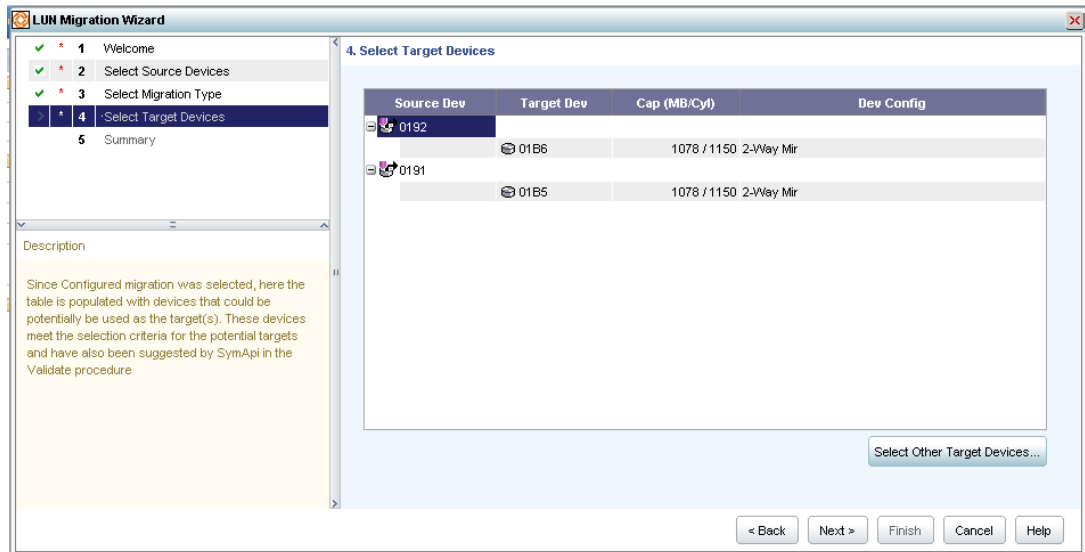
Configuration with Symmetrix Management Console - 54

In this step the migration type is specified – Configured or Unconfigured. For either type of migration one has to also choose the desired Protection Type (this will be the protection at the end of the migration) and the Disk Group. In this example we have chosen Configured devices with a protection of RAID1 (2-Way-Mir) which are located in Disk Group 2.

Click Next to proceed.

For migration to Unconfigured space the next step would be to click on the Finish button.

LUN Migration Wizard – Select Target Devices



© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 55

For Migration using configured devices the wizard automatically picks appropriate target devices. The selection can be changed by clicking on the “Select Other Target Devices”. In this example devices 1B5 and 1B6 have been chosen by the Wizard.

Click Next to continue.

LUN Migration Wizard – Summary

LUN Migration Wizard

5. Summary

Step Name

3 Select Migration Type

Migration Type : configured
Target Disk Group : 2
Migration Protection : RAID1

4 Select Target Devices

Source Dev	Target Dev	Cap (MB/Cyl)	Dev Config
0192	01B6	1078 / 1150	2-Way Mir
0191	01B5	1078 / 1150	2-Way Mir

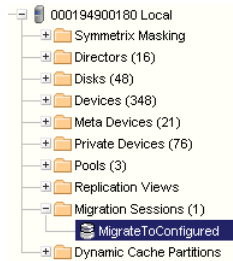
< Back Next > **Finish** Cancel Help

Progress Monitor

Execution of Migration in Progress

Review the Summary and click Finish to execute the LUN Migration.

Migration Session - Properties



Properties - 000194900180 Local/MigrateToConfigured

General Session Pairs (2)

Session Name	MigrateToConfigured
Invalid Tracks	0
Status	MigrInProg
Percent Complete	100

Properties - 000194900180 Local/MigrateToConfigured

General Session Pairs (2)

Src	Tgt	Invalid Tracks	Status	% Done	Tgt Disk Group	Tgt Protection
0191	01B5	0	Migrated	100.02		RAID-1
0192	01B6	0	Migrated	100.02		RAID-1

The LUN Migration Wizard creates the migration session. The properties view of the migration session can be used to monitor the session. Once the session is migrated it can be terminated.

Terminate Migration Session

Migration >

Terminate Migration Session ...

Question

? You are about to terminate Migration Session MigrateToConfigured, do you wish to continue?

Yes No

- Configuration after migration

Devices 191 and 192 are RDF1+Mir after migration

Properties - SMC DemoDG [900180] Standard Devices (2)				
Dev	LDev	Dev Config	Cap (MB/Cyl)	Dev Status
0191	DEV001	RDF1+Mir	1078 / 1150	Ready
0192	DEV002	RDF1+Mir	1078 / 1150	Ready

Devices 1B5 and 1B6 are RAID-6 (6+2) after migration

Properties - 000194900180 Local/Unmapped (10)						
Dev	Dev Config	Cap (MB/Cyl)	Dev Status	Reserved	Emulation	# Paths
01B6	RAID-6 (6 + 2)	1078 / 1150	Ready	No	FBA	0
01B5	RAID-6 (6 + 2)	1078 / 1150	Ready	No	FBA	0

© 2009 EMC Corporation. All rights reserved.

Configuration with Symmetrix Management Console - 58

Once the migration has completed the migration session should be terminated. To terminate a migration session, right click on the session and choose Migration → Terminate Migration Session.

In this example devices 191:192 are RDF1+Mir (SRDF RAID1 protected) after the migration. Whereas devices 1B5:1B6 are now RAID-6 (6+2) devices.

Demonstration

- [05 LUN Migration using unconfigured space](#)

Click on link to view demo.

In this demonstration two SRDF R1 RAID6 devices in Disk Group 1 are migrated to unconfigured space on Disk Group 2 still keeping the same RAID6 protection.

Module Summary

Key points covered in this module:

- Setting Symmetrix and Symmetrix Port attributes
- Configuring Symmetrix devices
- Managing Device pools
- Allocating storage to hosts with Autoprovisioning groups
- Configuring SRDF groups and RDF pairs
- Performing LUN migration

These are the key points covered in this module. Please take a moment to review them.