

EMC Solutions Enabler 7.0 Overview and Update

Don Fried-Tanzer

EMC WORLD
ORLANDO 2009

Objectives



At the end of this session, you will be able to

- Briefly explain the functionality enhancements provided by Solutions Enabler 7.0
- Describe how Solutions Enabler can simplify your storage management processes with enhancements to **symconfigure**, **symmask** and the new **symaccess** command
- Describe how Solutions Enabler 7.0 extends your replication options with enhancements to **symclone**, **symrdf**, and **symrcopy**

Agenda



- Storage Management Enhancements
 - Auto-provisioning Groups, **symaccess**
 - Symmetrix DMX Masking enhancements, **symmask**
 - Enhanced Virtual LUN Technology, **symmigrate**
 - Configuration changes, **symconfigure**
- SRDF Enhancements, **symrdf**
 - Add/remove devices to/from an active SRDF/A session
 - SRDF/Extended Distance Protection (SRDF/EDP)
 - SRDF/Star, **symstar**
- TimeFinder Enhancements
 - TimeFinder/Mirror emulation enhancements
 - Clone from clone target, **symclone**, **symmir** (emulation)
 - Command Scope, **symclone**, **symsnap**, **symmir**
 - Snap Recreate, **symsnap**
- Open Replicator for Symmetrix Enhancements, **symrcopy**
 - TimeFinder/Snap source for cold push
 - Support for AS400 (iSeries) devices pull from a remote Symmetrix system

Voice of Customer Drives Action to Deliver Result



Customer Feedback:

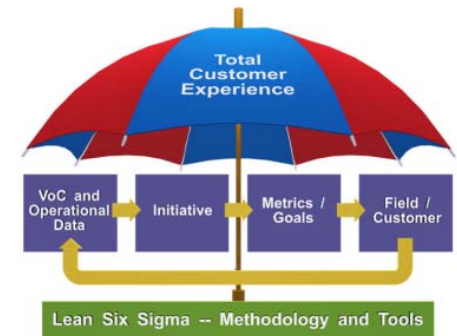
1. Device masking takes too many steps
2. We want to migrate to a different storage tier and change RAID protection at the same time
3. Storage configuration could be easier
4. We want to move new devices into SRDF/A without suspending it
5. Replication operations should be restricted to the group we are using

Results:

1. Fast and easy device masking
2. Change storage tier and RAID protection at same time
3. Simpler and more concurrent configuration change
4. No disaster recovery outage moving new devices into an SRDF/A group
5. Simpler and more logical replication operations working with groups

EMC's Action:

1. Auto-provisioning, **symaccess**
2. Enhanced Virtual LUN Technology, **symmigrate**
3. Concurrent configuration changes, richer command line, **symconfigure**
4. Enginuity Consistency Exempt, **symrdf movepair -cons_exempt**
5. Command Scope, **SYMAPI_COMMAND_SCOPE=ENABLE**



Auto-provisioning Groups Versus Individual Masking



- Why Auto-provisioning?
 - **symmask** masks individual initiators to a single front end director port
 - Small granularity more difficult to scale in increasingly more common environments
 - Storage arrays with over 100K devices
 - Large cluster requirements
 - Virtual machine and server farms
- **symaccess**, Auto-provisioning Groups
 - Enhance storage provisioning ease-of-use for customers
 - **IG**, Initiator Group, all the initiators that have common resources needed
 - **SG**, Storage Group, the devices that will be shared across the IG
 - **PG**, Port Group, the directors and ports that will have the masking applied to them
 - **MV**, Masking View, the association of the IG, SG, and PG that make up the actual masking as seen from the host
 - Easy to update, for example adding a device to a SG automatically makes it visible to all initiators (IG) on all ports (PG) in the view (MV)
- Benefit: Masking multiple initiators in fewer steps

Create the Storage, Port, and Initiator Groups



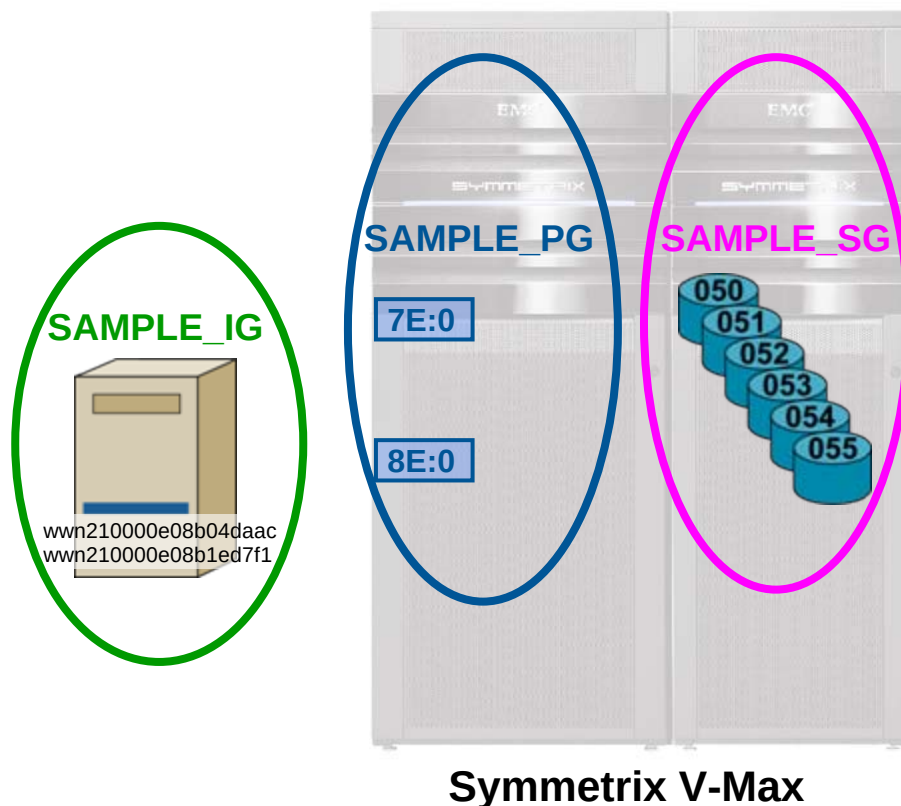
- Create the Storage Group (**SG**)

```
symaccess -sid 234  
-name SAMPLE_SG -type storage  
create devs 050:055
```
- Create the Port Group (**PG**)

```
symaccess -sid 234 create  
-name SAMPLE_PG -type port  
create 7E:0,8E:0
```
- Create an empty Initiator Group (**IG**)

```
symaccess -sid 234  
-name SAMPLE_IG  
-type initiator create
```
- Populate the **IG** with both host HBAs

```
symaccess -sid 234  
-name SAMPLE_IG  
-type initiator  
-wwn 210000e08b04daac add  
symaccess -sid 234  
-name SAMPLE_IG  
-type initiator  
-wwn 210000e08b1ed7f1 add
```

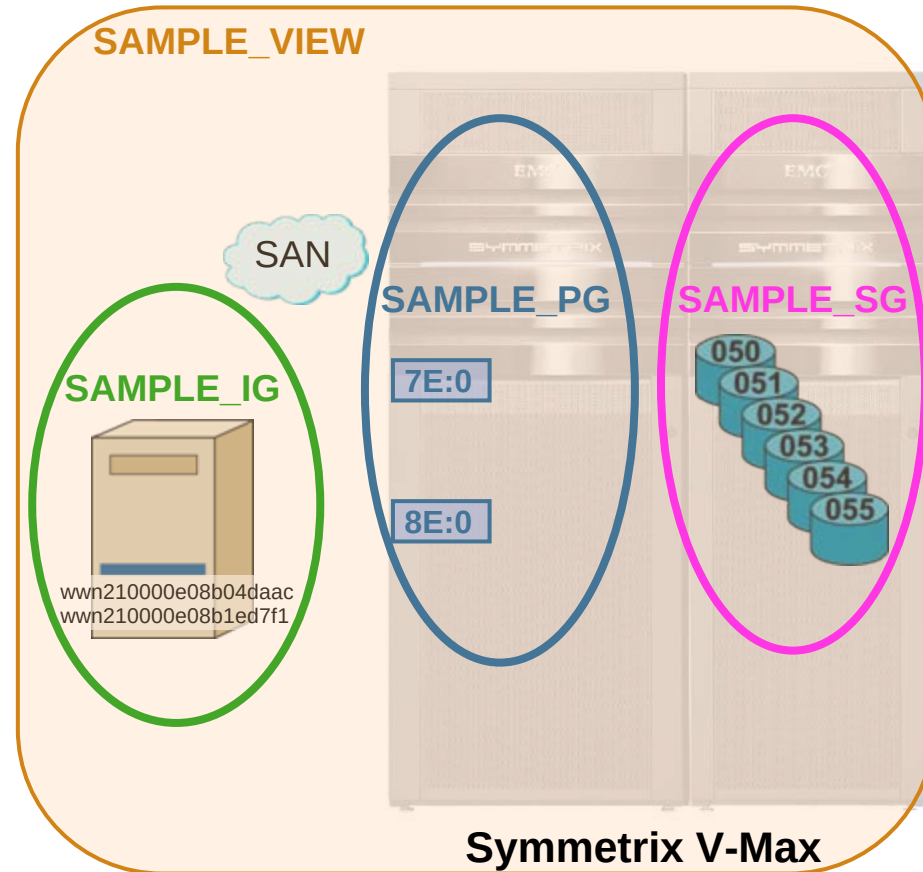


Masking View (MV)

- Link the SG, PG, and IG together into a MV, making the chosen devices visible on the host

```
symaccess -sid 234  
create view -name SAMPLE_VIEW  
-sg SAMPLE_SG -pg SAMPLE_PG  
-ig SAMPLE_IG
```

- During MV creation unmapped devices will automatically be mapped to the ports in the PG
 - Functionality added in response to requests
 - Will extend time to complete masking operation
 - Also applies when unmapped devices are added to an existing MV



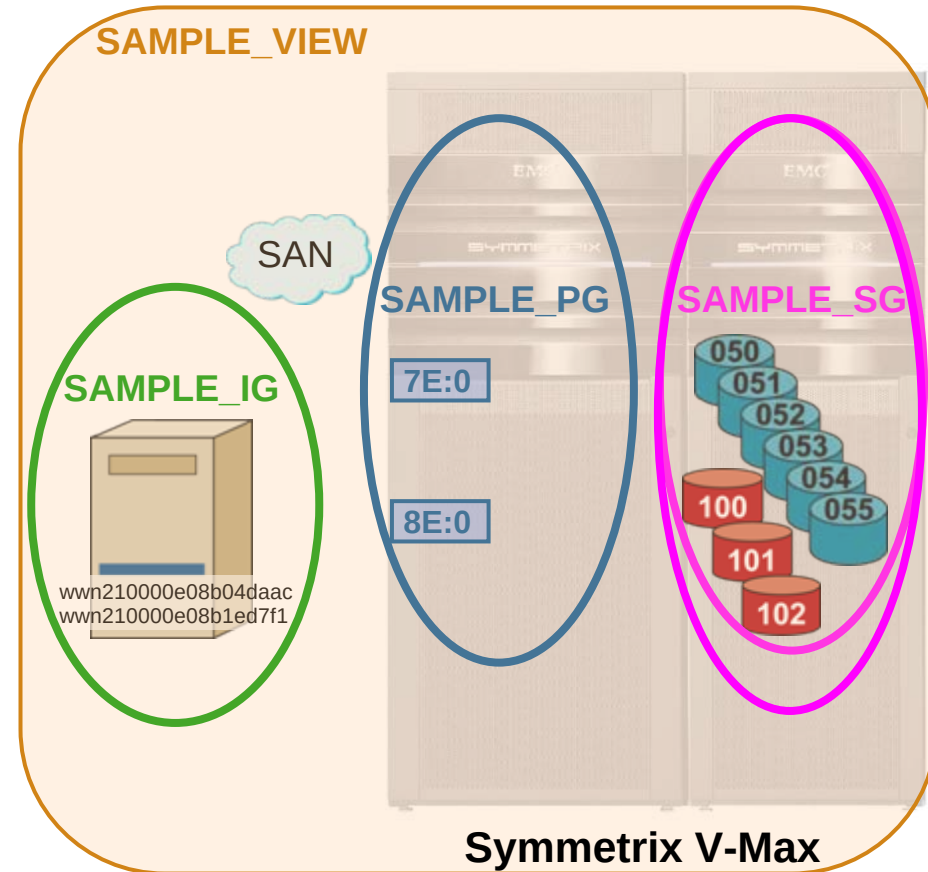
Devices Added to a Storage Group Automatically Add to the Masking View



- Adding devices 100-102 to the existing SG makes them visible to all of the same host initiators through the same V-Max ports

```
symaccess -sid 234 -name SAMPLE_SG  
-type storage add devs 100:102
```

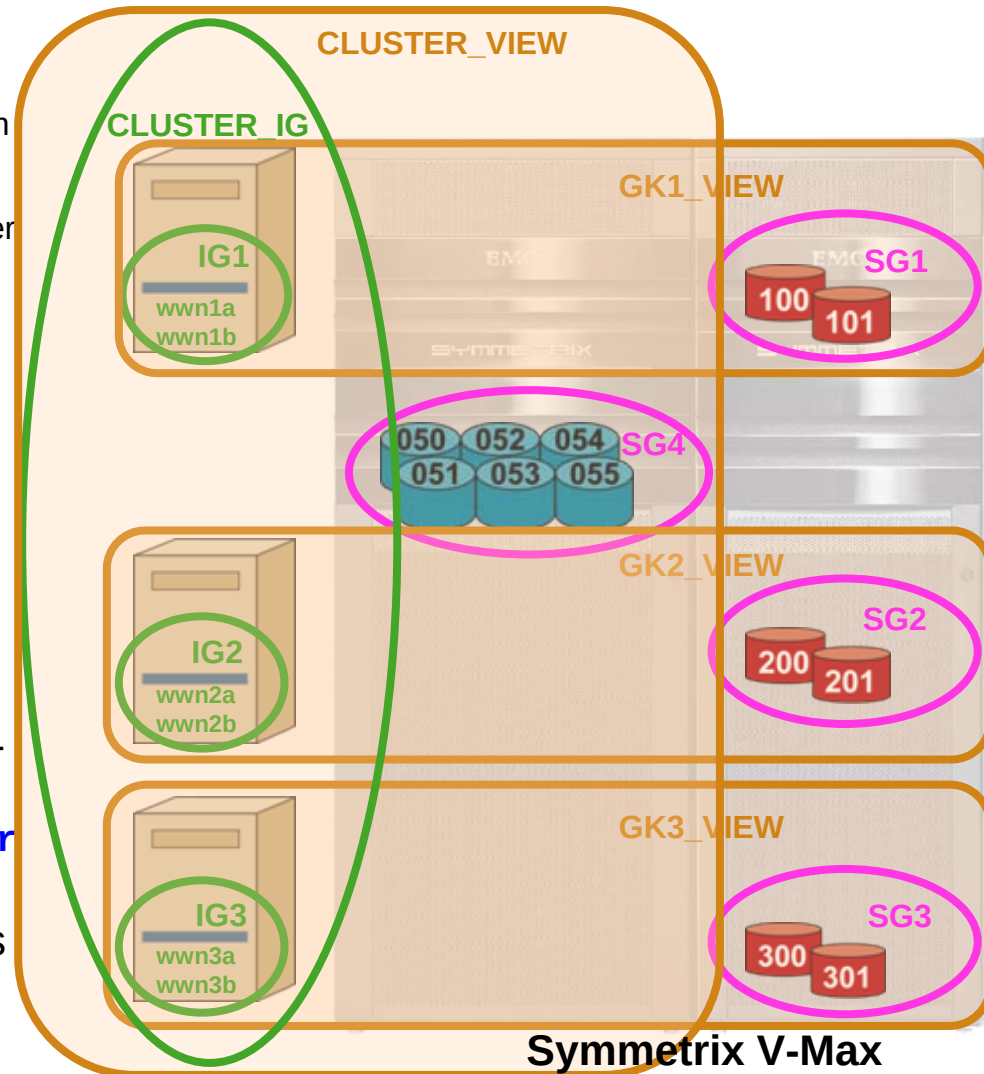
- In a similar fashion, if the existing devices needed to be masked to a different port or to more initiators, these elements can be added/deleted from their respective groups



Cascaded Initiator Group

- In a clustered environment
 - Some devices (e.g. gatekeepers) need to be seen only by individual hosts in the cluster
 - SG1: 100,101; SG2: 200, 201; SG3: 300, 301
 - Most devices need to be seen by the entire cluster
 - SG4: 050:055
- Initiator groups per host in cluster
 - IG1: wwn1a, wwn1b; IG2: wwn2a , wwn2b;
IG3: wwn3a , wwn3b
- Single shared port group
 - PG1: 7E:0, 8E:0 (not shown)
- Three host specific views (gatekeepers)
 - GK1_VIEW: IG1, PG1, SG1
 - GK2_VIEW: IG2, PG1, SG2
 - GK3_VIEW: IG3, PG1, SG3
- Cascaded Initiator Group for whole cluster

```
symaccess create -sid 234
-name CLUSTER_IG -type initiator
-ig IG1, IG2, IG3
```
- Make all cluster devices visible to all hosts
 - CLUSTER_VIEW : CLUSTER_IG, PG1, SG4



Symmetrix DMX Masking Alias Enhancements



- When devices are masked on several ports of the Symmetrix DMX to all the initiators of a host the number of **symmask** commands required to map the devices = # of initiators × # of ports
- Aliasing associates human-readable names with initiators
 - symmask -sid <SymmID> -wwn <wwn> rename <awwn>**
 - **symmask** can use the alias name instead of the WWN, in a number of actions
 - awwn <awwn>**
- By convention, an alias name is often two parts: **hostname_adapter**
 - Therefore multiple aliases on this same host would have a common **hostname** prefix
- SE 7.0 will execute a single **symmask** command against multiple initiators that have aliases that share the same prefix
 - symmask -sid <SymmID> -awwn <common_awwn_prefix> . . .**

Enhanced Virtual LUN Technology



- New CLI command **symmigrate**
 - Symmetrix volumes can be migrated between disk and/or protection types
 - Standard volumes only – not Thin Devices (TDEVs)
- Supports use of both configured and unconfigured disk space
 - Unconfigured disk space with no hyper assigned
 - Configured symdevs not currently in use (not mapped)
- Minimal impact
 - Active replication (TimeFinder, SRDF) on symdevs being migrated stays intact
 - Configuration lock only held at times bin file is being changed (rather than for duration of entire migration)
- VLUN migrations managed as sessions
 - Each session defines a group of Symmetrix volume(s) being migrated
 - Maximums
 - 4 concurrent active sessions (validate, establish, terminate) can be run concurrently
 - Up to 128 migrating devices per Symmetrix V-Max Engine
 - 16 concurrent active and passive (list, query) sessions

Enginuity 5874 Mirror Position Changes



- Number of mirror positions no longer an issue
 - All RAID types now presented as a single mirror (RAID Virtual Architecture)
 - Mirrors now represent an entire RAID group for all protection types
 - Permanent member sparing used instead of Dynamic sparing
 - TimeFinder/Clone emulation used instead of native TimeFinder/Mirror
- During a migration, a symdev can have both a primary and secondary RAID group
- CLI changes related to display of RAID groups
 - `symdev list`
 - `symdev show`
 - `syndisk show`

CLI Changes Related to Display of RAID Groups



symdev list, filter options added

- sec_raid, -sec_unprotected, -sec_raid1
- sec_raid5 [-sec_protection <3+1 | 7+1>]
- sec_raid6 [-sec_protection <6+2 | 14+2>]
- unprotected, -raid1

symdev list -rg option added

- Briefly list information about both primary and secondary RAID groups

symdev show

- Added line displays the Number of RAID Groups attached to the device
- New RAID Group Information section provides info about primary and secondary RAID groups
 - Use **-mode V65** to display old **RAID 5/RAID 6 Device Information** sections
- The following sections now contain a single entry per RAID group attached to device
 - **Back End Disk Director Information, Mirror Set Type**, Mirror Set DA Status, Mirror Set Inv. Tracks
- RAID-1 hyper entries in **Back End Disk Director Information** section are abstracted with details in new **RAID Group Information** section

symdisk show

- **RAID-1** hyper Type is now used instead of **Data** to represent RAID 1 hypers
- New **Mbr** column added to list the RAID group member number

symconfigure Changes



- Significant changes internally to allow
 - More than one non-conflicting **symconfigure** sessions to run concurrently
 - **symconfigure query** now displays multiple sessions and each session listing is preceded by a Session ID number
 - **symconfigure abort** new optional argument **[-session_id <session_num>]**
 - SE 7.0 blocks explicit release of locks held by a configuration change session on 5874
 - Richer single change: **create** + **map** + **set device attribute**
 - Underlying infrastructure changed, will help improve speed in subsequent releases
- For Enginuity 5874 **prepare** action is the same as **preview**
- Increased limits
 - 4 times bigger maximum device size, 262668 cylinders, ~240 GB
 - With 255 member maximum, FBA metas can now be ~60 TB
 - 512 maximum number of hypers per physical disk drive
 - Can use smaller hypers without wasting physical disk drive space
 - CKD-3380 supported (can have a maximum size of 3993 cylinders)

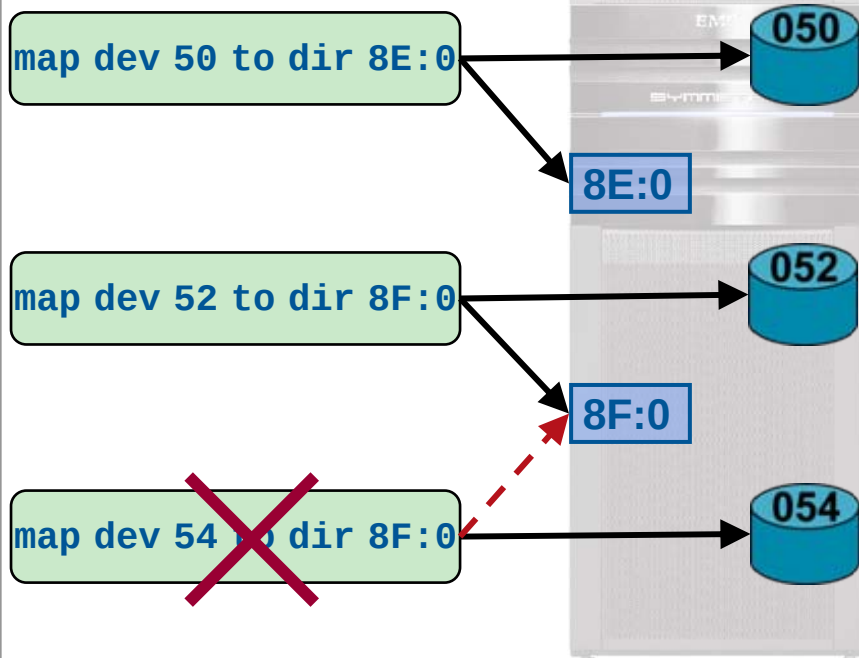
symconfigure No Longer Used For



- Masking; instead **symaccess** masking command can do mapping
 - Also, devices that are part of a masking view cannot be unmapped
- Changing local device mirror protection; instead use **symmigrate**
- Obsolete attributes for Enginuity 5874
 - Symmetrix attributes
 - **dynamic_rdf, concurrent_rdf, dynamic_concurrent_rdf**
 - **fba_multi_access_cache**
 - **max_hypers_per_disk**
 - Device attributes:
 - **RDB_Cksum**
 - Port attributes
 - **HP3000_Mode, Sequent, Siemens, Sunapee**
 - **VCM_State**
- New Symmetrix port attributes for Enginuity 5874
 - **ACLX**

Concurrent Configuration Change Sessions

Symmetrix V-Max



- DMX config change sessions were exclusive (Lock 15)
- 5874 will allow parallel execution of up to 4 configuration change sessions
 - Locks are finer grained (locks only the resources needed for the change types)
 - Config lock, locks the IMPL file
 - One per Symmetrix V-Max
 - Director locks
 - 1 lock per director
 - Front-end device lock
 - 1 per device
 - Back-end device lock
 - 1 per device
 - Locks are taken for a shorter duration
 - Conflicts can occur

Locks Needed for Configuration Actions



	Device Front End Lock	FBA Director Lock	Device Back End Lock	Static Config Lock
FBA Map / Unmap	yes	yes	no	no
Create Meta	yes	yes*	yes	no
Dissolve Meta	yes	yes*	yes	no
Add member	yes	yes*	yes	no
Delete Member	yes	yes*	yes	no
Meta Expand	yes	yes*	yes	yes
Create Device	yes	no	yes	yes
Create & Map in one api cmd.	yes	yes	yes	yes
Delete Device	yes	no	yes	yes
Change FBA Host Type	yes	yes	no	yes
Add / Delete Static RDF mirror	yes	no	no	yes
Permanent Member Sparing	no	no	yes	yes
Optimizer Swap	no	no	yes	yes
LUN Migration	no	no	yes	yes
Set / Unset DR1 / DR2 / DRX	yes	no	no	no
Change BCB/no BCB	yes	no	no	no
CKD Map / Unmap	no	no	no	yes
Change RDF Group attributes	no	no	no	no
Set Symmetrix attributes	no	no	no	yes

* Locked if the metahead is mapped to a director

Richer Creating Meta Devices Example



Old Way

Step 1:

```
create dev count=40,  
  size = 20000 cyl, emulation = fba,  
  config = 2-Way-Mir;
```

Step 2:

```
form meta from dev 600;  
add dev 601:603 to meta 600;
```

```
form meta from dev 604;  
add dev 605:607 to meta 604;
```

Create Dev (New Way)

```
create dev count=10,  
  size = 80000 cyl,  
  dynamic_capability = dyn_rdf  
  emulation = fba,  
  config = 2-Way-Mir,  
  mapping to dir 8G:0 starting lun=07,  
  mapping to dir 8H:0 starting lun=10,  
  device_attr = SCSI3_PERSIST_RESERV;
```

- Use case

- Create 10 (4-way) metas
- Map them to 2 ports
- Set the dynamic rdf capability
- Set scsi3_persist_resv

- Suppose Symmetrix is auto-meta ready

```
> symcfg list -sid 1153 -v
```

```
...  
Auto Meta : Enabled  
Minimum Auto Meta Size : 20000  
Auto Meta Member Size : 20000  
Auto Meta Configuration : Concatenated  
...
```

Virtual Provisioning Enhancements



- Data devices can be of any protection type
 - Snap/RDF using TDEVs bound to RAID-5 pools are now supported
- Delta Set Extension (DSE) with thin devices are now supported
- Draining of thin pool data devices fully supported
- Report written tracks on devices in thin pool

```
symcfg show -pool redsnapper -thin -detail -sid 111
```

```
Symmetrix ID: 000192600111
```

```
• • •  
Thin Devices(2):  
{
```

Sym Dev	Total Tracks	Pool Subs (%)	Pool Allocated		Pool Written		Status
			Tracks	(%)	Tracks	(%)	
01B0	69030	8	36396	4	28042	3	Bound
01B1	69030	8	40932	5	30886	4	Bound
Tracks	138060	17	77328	9	58928	7	

Agenda



- Storage Management Enhancements
 - Auto-provisioning Groups, **symaccess**
 - DMX Masking enhancements, **symmask**
 - Enhanced Virtual LUN Technology, **symmigrate**
 - Configuration changes, **symconfigure**
- **SRDF Enhancements, *symrdf***
 - Add/remove devices to/from an active SRDF/A session
 - SRDF/Extended Distance Protection (SRDF/EDP)
 - SRDF/Star, ***symstar***
- TimeFinder Enhancements
 - TimeFinder/Mirror emulation enhancements
 - Clone from clone target, **symclone**, **symmir** (emulation)
 - Command Scope, **symclone**, **symsnap**, **symmir**
 - Snap Recreate, **symsnap**
- Open Replicator for Symmetrix Enhancements, **symrcopy**
 - TimeFinder/Snap source for cold push
 - Support for AS/400 (iSeries) devices pull from a remote Symmetrix system

Enginuity 5874 Infrastructure Changes



- Support for 250 RDF Groups
 - No changes to any API or CLI externals
 - Group numbers 1 through 250 have been supported since 5772
 - Enginuity 5772 & 5773 could only define 128 of the 250 RDF Groups
 - Enginuity 5874 supports a maximum of 64 RDF Groups per RDF director
 - Enginuity 5772 & 5773 only supports a maximum of 32 RDF Groups per RDF director
- Support for 128 directors
 - Slices A, B, C, D, E, F, G, H
 - All CLIs that accept **-dir** or **-remote_dir** now accept directors **1A** – **16H** includes
 - **symrdf addgrp**
 - **symrdf modifygrp**
- Virtual Provisioning with SRDF now supports RAID 5
- Concurrent TimeFinder/Clone and SRDF Restore
 - Now allow a **symrdf restore** while a **symclone restore** is in progress

Move Devices In and Out of an Active SRDF/A Session



- Pre-Solutions Enabler 7.0
 - To add or remove a device to a SRDF/A session the user had to
 - Suspend all the devices in the SRDF/A session, making it inactive
 - Use **movepair** or **createpair** to add or remove the devices
 - **resume** the devices to make the session active again
 - A disaster recovery (DR) outage was required to perform this operation
 - After the **resume** following an add, the SRDF/A devices would be **SyncInProg** until all the invalids on all the devices in the SRDF/A session had cleared and 2 cycle switches occurred
- Enginuity Consistency Exempt Feature (5874 and Solutions Enabler 7.0)
 - Allows devices to be excluded from the dependent write consistency calculation
 - SRDF/A is a RDF Group level protocol
 - However, the Consistency Exempt attribute is maintained on a RDF Device level
 - Maintained by Enginuity on both the R1 and R2
 - Only the added devices will show as not **Consistent**
 - Is a persistent attribute until automatically removed by Enginuity when the
 - RDF pair is deleted (**deletepair**)
 - RDF pair is moved to another RDF Group (**movepair**)
 - Device is **resumed**, clears all invalid tracks, and 2 cycle switches occur

Consistency Exempt CLI Changes



- **symrdf -cons_exempt** command option
 - Available for actions
 - **createpair** – RDF mirrors become Consistency Exempt in the target RDF Group
 - **movepair** – RDF mirrors become Consistency Exempt in the target RDF Group
 - **suspend** – Devices become Consistency Exempt in the current RDF Group
 - Some operations now allowed on a subset of devices in an active SRDF/A session
 - Can **suspend** a subset with **-cons_exempt**
 - Can **resume** or **establish** devices that are in a Consistency Exempt state
- **symdev show**
 - In the **RDF Information** section, new **Device Consistency Exempt State** field showing either **Disabled** or **Enabled** (or possibly **Mixed** for meta head)
- **symrdf query**
 - New **MODES** flag
(Consistency) **E(xempt): X = Enabled, . = Disabled, M = Mixed**
- **symrdf list**
 - New **MODES** flag
(Consistency) **E(xempt): X = Enabled, . = Disabled, M = Mixed**
 - New **-cons_exempt** option to list all devices in a Consistency Exempt state

Add a Device to an Active SRDF/A Session

Example



- Best practice to create the new RDF pairs in a temporary RDF Group and synchronize them to minimize the transition to **Consistent** state

```
symrdf -file newpair -sid 50 -rdfig 118 createpair -type r1 -establish
symrdf -file newpair -sid 50 -rdfig 118 verify -synchronized
```
- Suspend the device so that it can be moved into the SRDF/A RDF Group

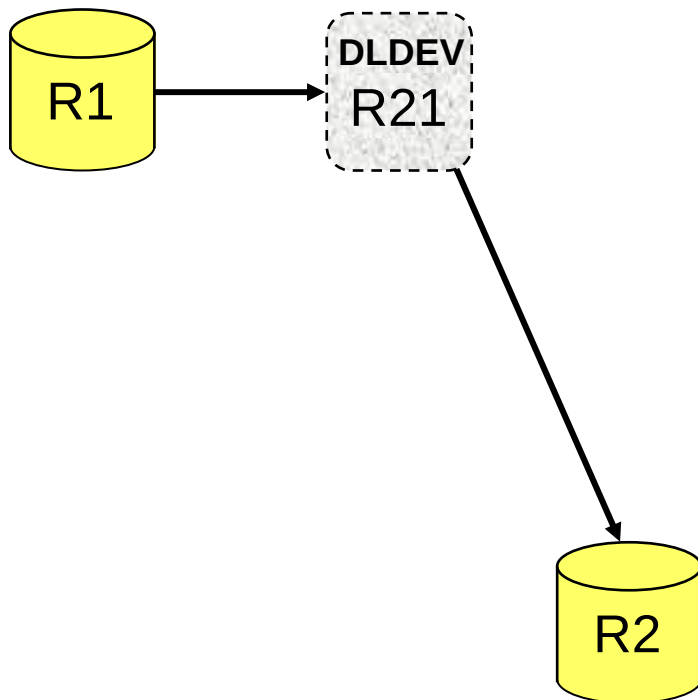
```
symrdf -file newpair -sid 50 -rdfig 118 suspend
```
- Move the device from the current RDF Group into the active SRDF/A RDF Group

```
symrdf -file newpair -sid 50 -rdfig 118 movepair -new_rdfg 18 -cons_exempt
```
- Resume the devices

```
symrdf -file newpair -sid 50 -rdfig 18 resume
```
- Wait for the devices to become consistent!

```
symrdf -file newpair -sid 50 -rdfig 18 verify -consistent
```
- Only after **Consistent** state is reached is the device safe to start being used by the Application

SRDF/Extended Distance Protection (SRDF/EDP)



- Extension of Cascaded SRDF
 - Lower cost solution due to reduced hardware, power, and cooling
- New “Diskless” R21 device (DLDEV)
 - Device has no disk storage, cache storage of changes only
 - Secondary site cannot be used for restart/recovery
 - First hop stops replication if second hop lost
- Enginuity release requirements
 - DLDEV R21 must be at 5874 or higher
 - R1 and R2 can be at 5773 or higher
- Creating DLDEV R21 devices
 - **create dev** command using **DLDEV** for the **device_configuration**
 - For static SRDF can also use **RDF1-DLDEV**, or **RDF2-DLDEV**
 - Same concurrent RDF config, using either
 - **symrdf createpair**
 - **symconfigure -cmd add rdf mirror**

SRDF/EDP – SRDF Feature



- Diskless devices are only usable when they are R21 devices
 - R1 → DLDEV-R21 → R2
 - Can be R1, R2, R11, and R22 during configuration
- All SRDF Control/Set/Query/Checkpoint/Verify commands are supported
 - Some restrictions apply on controls
 - Enginuity requires the R21 → R2 relationship to be made RW on the link first
 - Many SRDF controls are blocked for a R1 → R2 relationship where one device is diskless (e.g., no failover)
- Restrictions exist on the SRDF modes that are supported

R1 → R21	R21 → R2
Synchronous	Asynchronous
Adaptive Copy Disk	Asynchronous
Adaptive Copy WP	Asynchronous
Synchronous	Adaptive Copy WP
Adaptive Copy Disk	Adaptive Copy WP
Adaptive Copy WP	Adaptive Copy WP

SRDF/EDP – CLI Changes



- **symdev show**
 - New DLDEV device configuration values
DLDEV, RDF1+DLDEV, RDF2+DLDEV, RDF21+DLDEV
 - **SA Status** always **N/A** since DLDEVs cannot be mapped to host
 - No local storage member
 - New field: **Paired with a Diskless Device** can be **True** or **False**
- **symrdf list -diskless_rdf**
 - Displays devices that are either Diskless RDF or paired with a Diskless RDF device
- **symdev list -dldev**
 - Used to list all the Diskless devices that are configured on a Symmetrix

SRDF Composite Group (CG) Changes



- CG functionality now compatible with DG functionality
 - The following **symrdf -cg** actions have been added
 - **deletepair**
 - **half_deletepair**
 - **swap**
 - **half_swap**
 - **failover -establish**
 - Actions are allowed on the following device lists
 - **-hop2, -bcv, -bcv -hop2, -rbcv, -brbcv**
 - Actions are blocked that will cause CG to become invalid
 - Actions that change the Device Config on a subset of devices in any device list
 - Actions on LDEVs
- SRDF/S Consistency is now only supported via RDF-ECA
 - The **-ppath** option is no longer valid on the following commands
 - **symcg create**
 - **symcg import** and **symcg importall**
 - **symcg list**
 - **symdg dg2cg**
 - **symrdb rdb2cg**
 - **symrdb tbs2cg**
 - **symvg vg2cg**

Selected SRDF/Star Enhancements

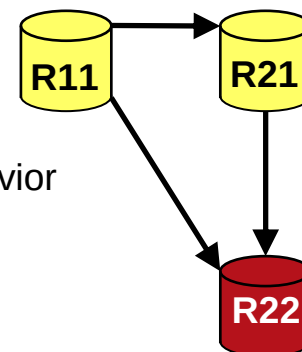


- Automatic distribution of Star internal definition file
 - The internal definition file is now saved on SFS and is automatically distributed to SFS on all three sites
 - SE 7.0 feature, not Enginuity dependent
 - Upgrade existing configurations with **symstar setup -distribute**
- SRDF/EDP support
 - Diskless R21 devices at Synchronous target site (*SiteB*) primarily supported for Cascaded Star mode
 - **symstar show** has a new **Diskless Device Site** field
- Concurrent R2 support

SRDF/Star Concurrent R2 Support



- Concurrent R2 devices at Asynchronous target site
 - Increased resiliency of Star commands
 - Improved performance
 - Supports all existing **symstar** actions, no change in external command behavior
 - Requires new **-opmode** option with **symstar setup** with possible values: **cascaded** or **concurrent**
- Upgrade existing configuration to use R22s
 - New **configure** action, example
`symstar -cg r22cg configure -add recovery_rdf_pairs -opmode cascaded`
- Concurrent R2 (R22) device has two SRDF mirrors
 - Device configuration value still shows as an RDF2 device
 - Only one SRDF mirror can be active on the link at a time
 - Like R11 displays in the CLI
 - **symdev show** with two **R1 Remote** mirrors and two **RDF Information Sections**
 - **symrdf -concurrent** now will display R22s
 - New **symrdf list -all** option displays all SRDF devices, with 2 lines for every Concurrent device (R11, R21 or R22)
 - Same concurrent SRDF configuration, adding a second R2 mirror to an existing R2 device, using either **symrdf createpair** or **symconfigure -cmd add rdf mirror**



Agenda



- Storage Management Enhancements
 - Auto-provisioning Groups, **symaccess**
 - DMX Masking enhancements, **symmask**
 - Enhanced Virtual LUN Technology, **symmigrate**
 - Configuration changes, **symconfigure**
- SRDF Enhancements, **symrdf**
 - Add/remove devices to/from an active SRDF/A session
 - SRDF/Extended Distance Protection (SRDF/EDP)
 - SRDF/Star, **symstar**
- TimeFinder Enhancements
 - TimeFinder/Mirror emulation enhancements
 - Clone from clone target, **symclone**, **symmir** (emulation)
 - Command Scope, **symclone**, **symsnap**, **symmir**
 - Snap Recreate, **symsnap**
- Open Replicator for Symmetrix Enhancements, **symrcopy**
 - TimeFinder/Snap source for cold push
 - Support for AS/400 (iSeries) devices pull from a remote Symmetrix systems

No TimeFinder/Mirror Native Support in Enginuity 5874



- BCV devices remain
- TimeFinder/Clone and TimeFinder/Clone Emulation remain
- TimeFinder/Mirror on legacy code (pre-5874) is still supported
- Main reasons
 - Local Replication Roadmap
 - New “contradicting” features:
 - Block CRC
 - Raid Virtual Architecture

TimeFinder/Clone Enhanced Features



	TimeFinder/Mirror	TimeFinder/Clone
Full Volume Replication	Yes	Yes
High Performance	Yes	Performance improvements in this release
Dynamic Protection	Yes	Leverage online configuration changes
Instant Target Data Guarantee	Yes	No
Optimizer	Supporting technology	Enhanced Virtual LUN is supporting technology
Meta Device Reconfiguration	Supporting technology	Clone with duplicate writes is supporting technology
SRDF R2 to R1 Restore from Target	Yes	Yes, new feature
Cascading Target Images	Yes	Yes, new feature

Clone from Clone Support



- Enginuity version 5874 and higher
- Clone A→B→C
 - Both sessions can be **Created**
 - Both sessions can be **Precopy**
 - First session can be **created** or **CopyInProg** when second **Precopy**
 - Cannot both be **CopyInProg** at the same time
 - Target device cannot be target for second copy (A→B copied, B→C restore)
- Emulation supports only operations supported by TimeFinder/Mirror native
 - Clone operations supported as clone from clone
 - Clone A→B to emulation B→C (C must be BCV)
 - Emulation A→B to clone B→C (B must be BCV)
- Enhancement is for incremental support, full copy clone-from-clone is available in earlier releases

Caveats with Clone Compared to TimeFinder/Mirror



- Clone performance hints
 - Use **-precopy** to limit Copy on First Write (COFW) penalty
 - Use Dynamic Cache Partitioning to limit Clone cache use
 - With smallest 10% partition, cache copy rate affect minimal to none
 - Cache usage drop from 50% of all cache to 50% of partition size
- Clone does not support
 - TimeFinder/Mirror-like **restore** operation, all Clone restores are Protected Restores
 - Clone does not support rarely used Reverse Split
- Clone target is not guaranteed to have all data at **activate** time
- Cascaded Clone requires **terminate** before a **restore**
- Over time, probably a best practice to eventually stop using TimeFinder/Clone Emulation (**symmir**) and convert to native TimeFinder/Clone scripts (**symclone**)

Command Scope in SE 7.0



- No Enginuity dependency
- Command Scope can be enabled or disabled with
 - Environmental variable **SYMCLI_COMMAND_SCOPE**
 - Options file setting **SYMAPI_COMMAND_SCOPE = ENABLE**
 - Default is disabled, so behavior remains unchanged
- Enabling Command Scope
 - Limits command actions to specified group
 - Takes no action on "paired" TimeFinder devices outside the group
 - Query/Verify commands also only report on grouped devices

Command Scope Example `syndg show`



`syndg show TestDg`

Standard (STD) Devices (2):

{

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)
DEV001	N/A	0101		RW	2063
DEV002	/dev/sdeu	0102		RW	2063

}

BCV Devices Locally-associated (2):

{

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)
BCV001	N/A	0201		NR	2063
BCV002	N/A	0202		NR	2063

}

Command Scope Disabled Examples



```
symmir -g TestDg query -multi
```

Standard Device			BCV Device		State	
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0101	0	BCV001	0201 *	0	Synchronized
DEV002	0102	0	N/A	003F	0	Synchronized

```
symmir -g TestDg verify -synched
```

All devices in the group 'TestDg' are in 'Synchronized' state.
Not All of the Paired Target Devices are associated with the group 'TestDg'

(*): The paired BCV device is associated with this group.

```
symmir -g TestDg establish -full -v -noprompt
```

```
SELECTING Source devices in the group:
```

```
Device: 0101 [SELECTED]
```

```
Device: 0102 [NOT SELECTED - ALREADY IN BCV STATE]
```

```
Device is already in this BCV state
```

Command Scope Enabled Examples



```
symmir -g TestDg query -multi
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0101	0	BCV001	0201 *	0	Synchronized
. . .						

```
symmir -g TestDg verify -synched
```

Not All devices in the group 'TestDg' are in 'Synchronized' state.

* DEV002 does not have a session.

Command Scope Enabled Examples (cont.)



```
symmir -g TestDg establish -full -v
```

'Full Establish' operation execution is in progress for device group 'TestDg'. Please wait...

SELECTING Standard devices in the group:

Device: 0101 [SELECTED]

Device: 0102 [SELECTED]

SELECTING BCV devices associated with the group:

Device: 0201 [SELECTED]

Device: 0202 [SELECTED]

PAIRING of Standard and BCV devices:

Devices: 0101(S) - 0201(B)

Devices: 0102(S) - 0202(B) [CONCURRENTLY PAIRED]

STARTING a BCV 'ESTABLISH' operation.

The BCV 'ESTABLISH' operation SUCCEEDED.

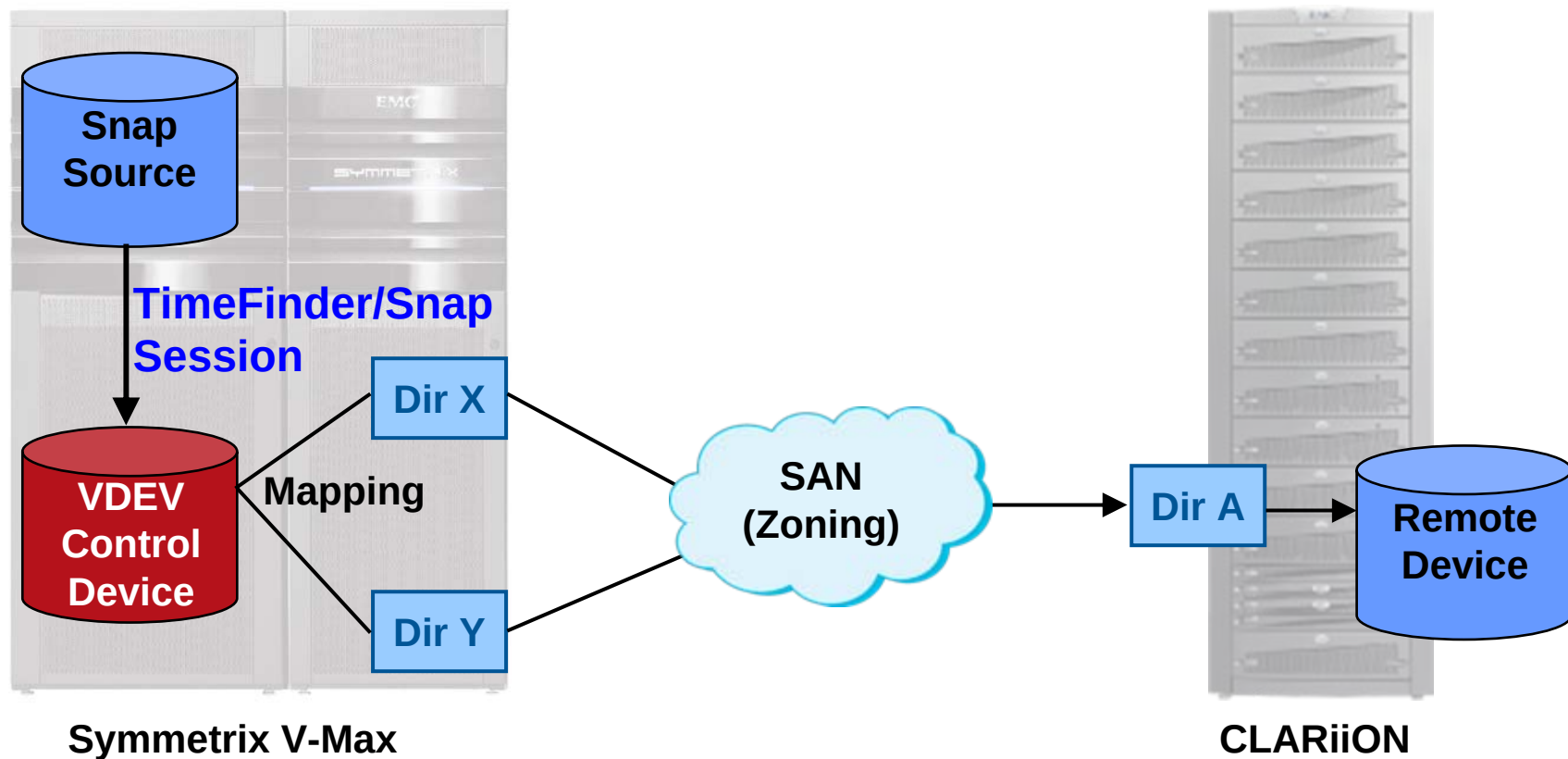
'Full Establish' operation successfully initiated for device group 'TestDg'.

Agenda



- Storage Management Enhancements
 - Auto-provisioning Groups, **symaccess**
 - DMX Masking enhancements, **symmask**
 - Enhanced Virtual LUN Technology, **symmigrate**
 - Configuration changes, **symconfigure**
- SRDF Enhancements, **symrdf**
 - Add/remove devices to/from an active SRDF/A session
 - SRDF/Extended Distance Protection (SRDF/EDP)
 - SRDF/Star, **symstar**
- TimeFinder Enhancements
 - TimeFinder/Mirror emulation enhancements
 - Clone from clone target, **symclone**, **symmir** (emulation)
 - Command Scope, **symclone**, **symsnap**, **symmir**
 - Snap Recreate, **symsnap**
- **Open Replicator for Symmetrix Enhancements, **symrcopy****
 - **TimeFinder/Snap source for cold push**
 - **Support for AS/400 (iSeries) devices pull from a remote Symmetrix systems**

TimeFinder/Snap (VDEV) Control Device for Open Replicator Cold Push



Required Order of **symsnap** and **symrcopy** Commands



- Create the snap session
`symsnap -f snap -sid 111 create -noprompt`
- Create the Open Replicator session
`symrcopy -f ors -sid 111 create -diff -copy -cold -noprompt`
- Activate the SNAP session. This command creates the point-in-time
`symsnap -f snap -sid 111 activate -not_ready -noprompt`
- Activate the Open Replicator session. Open Replicator copy begins at this time
`symrcopy -f ors -sid 111 activate -noprompt`
- *Wait for Open Replicator session copy to complete*
- Recreate the snap session
`symsnap -f snap -sid 111 -nop recreate`
- Recreate the Open Replicator Session
`symrcopy -f ors -sid 111 -nop recreate`
- Activate the SNAP session. This command creates the point-in-time
`symsnap -f snap -sid 111 -nop activate -not_ready`
- Activate the Open Replicator Session. Open Replicator copy begins at this time
`symrcopy -f ors -sid 111 -nop activate`
- Terminate the Open Replicator session
`symrcopy -f ors -sid 111 -nop terminate`
- Terminate the snap session
`symrcopy -f snap -sid 111 -nop terminate`

Open Replicator Support for AS/400 (iSeries) Devices



- Hot and cold pull sessions to AS400 emulation devices
 - **donor_update** supported
- **symsan** will report WWN for remote AS400 devices
- Control device must be on Enginuity 5874 and higher
- Remote device must be on Symmetrix system

symSAN Command New Flags – Thin Device and Symmetrix Device



```
symSAN -sid 111 -sanluns list -wwn 5006048AD5F004CE -dir 8e -port 0
```

```
Symmetrix ID:      000192600111
Remote Port WWN:   5006048AD5F004CE
```

```
ST
A
T Flags
DIR:P E ICRTHS Block Size Capacity (MB) LUN Num Dev Num LUN WWN
-----
08E:0 RW ...F.X 512 6 0 004C 60060480000190300179533030303443
08E:0 RW ...F.X 512 2063 1 0012 60060480000190300179533030303132
08E:0 RW ...FXX 512 2063 2 0013 60060480000190300179533030303133
08E:0 RW ...FXX 512 2063 3 0014 60060480000190300179533030303134
08E:0 RW ...F.X 512 2063 4 0015 60060480000190300179533030303135
08E:0 RW ...F.X 512 2063 5 0016 60060480000190300179533030303136
```

Legend:

Flags: (I)ncomplete : X = record is incomplete, . = record is complete.
(C)ontroller : X = record is controller, . = record is not controller.
(R)eserved : X = record is reserved, . = record is not reserved.
(T)ype : A = AS400, F = FBA, C = CKD, . = Unknown
t(H)in : X = record is a thin dev, . = record is not a thin dev.
(S)ymmetrix : X = Symmetrix device, . = not Symmetrix device

Summary and Questions



- Briefly explained the functionality enhancements provided by Solutions Enabler 7.0
- Described how Solutions Enabler can simplify your storage management processes with enhancements to **symconfigure**, **symmask** and the new **symaccess** command
- Described how Solutions Enabler 7.0 extends your replication options with enhancements to **symclone**, **symrdf**, and **symrcopy**

Related Sessions at EMC World



- Symmetrix V-Max Series with Enginuity Overview
- Understanding New Features of Enginuity 5874 for Symmetrix V-Max
- Symmetrix Auto-provisioning Groups: Simplified Storage Implementation Using Symmetrix Management Console (SMC) and the Symmetrix CLI
- Performing In-Array Migrations using Symmetrix Enhanced Virtual LUN Technology
- Deep Dive - SRDF/Star for Open Systems Design and Implementation (Cascaded, Concurrent, SRDF/EDP)
- SRDF/A Best Practices
- Symmetrix Storage Provisioning Best Practices
- Symmetrix Device Masking Enhancements Hands-on Lab
- Enhanced Virtual LUN and Open Replicator Migration Hands-on Lab
- Solutions Enabler SRDF Hands-on Lab
- Hands-on with Symmetrix Virtual Provisioning
- Benefits of Scripting EMC TimeFinder/Clone Native Command Lines for Your Symmetrix Local Data Replication Operations

Related Technical Documentation



- *EMC Symmetrix V-Max Series Product Guide*
- *EMC Symmetrix V-Max EMC Enginuity Release Notes*
- *White Paper: New Features in EMC Enginuity 5874 for Symmetrix Open Systems Environments - A Detailed Review*
- *EMC Solutions Enabler Release Notes*
- *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*
- *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
- *EMC Solutions Enabler Symmetrix CLI Command Reference*
- *EMC Solutions Enabler Symmetrix Open Replicator CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*
- *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*

EMC²[®]

where information lives[®]