White Paper

symantec.
Confidence in a connected world.

EMC²
where information lives®

EMC Symmetrix V-Max
with Veritas Storage Foundation

Applied Technology

Abstract

This white paper details the benefits of deploying EMC® Symmetrix® V-Max™ Virtual Provisioning™ and Veritas Storage Foundation™ with SmartMove by Symantec.

May 2009

Table of Contents

Executive summary

EMC® Symmetrix® V-Max™ storage arrays continue to provide increased storage utilization and optimization, enhanced capabilities, and greater interoperability and security, as well as multiple ease-of-use improvements. The Symmetrix V-Max system is an innovative storage array built around a scalable fabric interconnect design. It incorporates powerful new multi-core processors and can seamlessly grow from an entry-level configuration into the world's largest storage system.

Virtual Provisioning™, generally known in the industry as "thin provisioning," enables organizations to improve ease of use for deploying and provisioning application storage. Virtual Provisioning can also enhance performance and increase capacity utilization for certain applications and workloads. The implementation of Virtual Provisioning for Symmetrix V-Max storage arrays directly addresses improvements in storage infrastructure utilization, as well as associated operational requirements and efficiencies.

The concepts of "thin friendliness" are increasingly important. While virtual or thin provisioning is designed to be transparent to the applications and hosts, the host stack, host file system, and host volume manager are not transparent to the virtually provisioned environment. Veritas Storage Foundation from Symantec provides substantial benefits for companies using or planning to use EMC Virtual Provisioning technologies.

As Virtual Provisioning support is released on most storage arrays in the market, there is a growing need to migrate from thick to thin storage. Traditional block-level migration tools copy both data and empty space from a thick storage device, resulting in a "thin" provisioning system that uses just as much storage as the legacy system.

This white paper addresses Veritas Storage Foundation with SmartMove technology, which allows online migration of data from thick to thin storage while automatically optimizing storage utilization and enabling customers to reclaim unused space from existing applications. SmartMove is available on UNIX and Linux in Storage Foundation 5.0 MP3, and on Windows in Storage Foundation for Windows 5.1. It is supported on any EMC (Symmetrix DMX™ or V-Max) storage array running Enginuity™ 5773.xx or later.

Introduction

A new challenge facing storage administrators today is the migration of data and deploying into virtually provisioned environments. In order to implement this solution, current systems, applications, and databases need to be migrated from traditional thick (also called "fat") storage devices onto thin devices.

This white paper addresses the implementation of Veritas Storage Foundation with EMC Virtual Provisioning technology to provide data migration capabilities. The ability to migrate data from traditional (or thick) storage onto thin devices will allow customers to take advantage of this new technology to increase storage utilization and optimization.

Audience

This white paper is intended for storage architects and administrators, and database administrators responsible for deploying Veritas Storage Foundation 5.0 MP3 with EMC Symmetrix DMX or V-Max using 5773 Enginuity.

Virtual Provisioning

Virtual Provisioning relies on the observation that most organizations over-allocate storage. When requesting capacity, many customers incorrectly estimate their storage needs to make sure they have enough capacity available in the future. Acquiring storage early in a project when funds are allocated is easier than purchasing it later, when there may be insufficient funds. The traditional approach is to pre-allocate and dedicate storage to a particular application or project, which may later prevent other applications from accessing this capacity, even if the amount allocated is not fully utilized.
Virtual Provisioning offers an alternative, allowing users to create a volume or file system of any size without allocating all of the physical capacity at the time of creation. Virtual Provisioning creates virtual disk drives, or logical unit numbers (LUNs) that appear to be one size, but whose actual physical storage only covers a fraction of their claimed size. If a LUN needs more storage, the storage array allocates more physical storage to it, without changing the presented size.
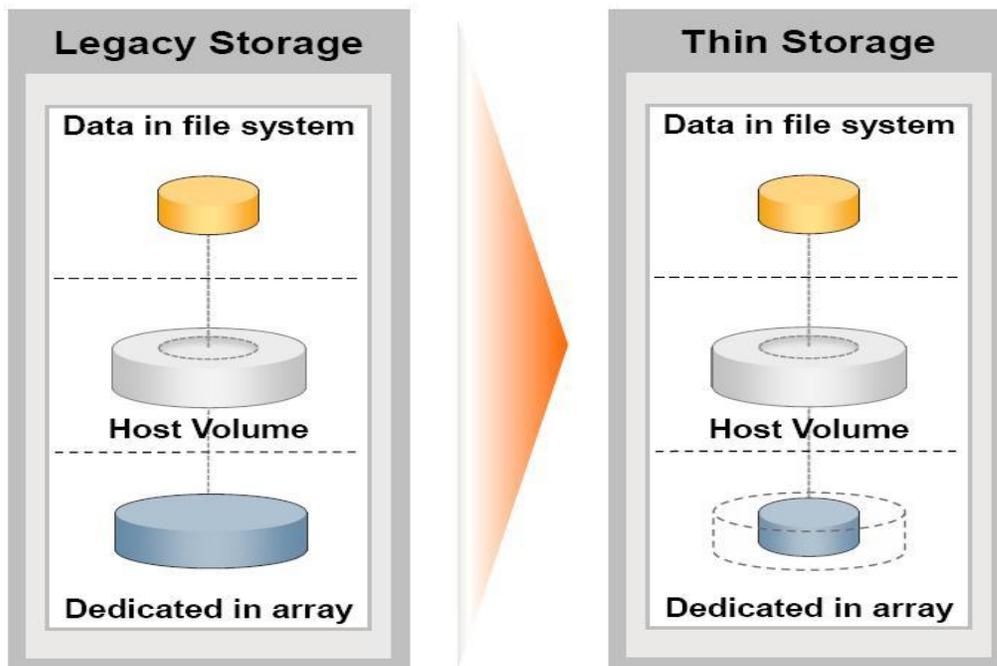
Figure 1. Traditional (thick) storage and thin provisioned storage utilization

The storage used in the thin scenario is the same as the file system (best case); actual storage usage can be higher, as later sections explain. Virtual Provisioning offers the following benefits over the traditional "thick" storage:

- Higher storage utilization rates

- Reduced capital expenditures as the storage utilization rates increase

- Reduced operational expenditures as Virtual Provisioning allows storage administrators to over-allocate with no upfront cost.

EMC Symmetrix V-Max Virtual Provisioning

EMC Symmetrix V-Max thin devices are logical devices that can be used in many of the same ways that traditional storage devices have been used. However, virtually provisioned thin devices do not need to have physical storage completely allocated at the time the device is created and presented to a host. The physical storage that is used to supply disk space to thin devices comes from a shared storage pool called a thin pool. The thin pool is comprised of devices called data devices that provide the actual physical storage to support the thin device allocations.

When a write is performed to a part of the thin device for which physical storage has not yet been allocated, the Symmetrix V-Max allocates physical storage from the thin pool for that portion of the thin device only. The Symmetrix V-Max operating environment, called Enginuity, satisfies the requirement by providing a block of storage from the thin pool called a thin device extent. This approach reduces the amount of storage that is actually consumed.

The minimum amount of physical storage that can be reserved at a time for the dedicated use of a thin device is referred to as the data device extent. The entire data device extent is physically allocated to the thin device at the time the storage allocation operation is performed. The data device extent is allocated from any one of the data devices in the associated thin pool.

The minimum amount of physical storage that must be mapped to a thin device at a time is called a thin device extent. For Symmetrix V-Max, the thin device extent size is the same as the data device extent size.

When a read is performed on a thin device, the data being read is retrieved from the appropriate data device in the thin pool to which the thin device is associated. If for some reason a read is performed against an unallocated portion of the thin device, zeroes are returned to the reading process.

When more physical data storage is required to service existing or future thin devices, for example, when a thin pool is approaching full storage allocations, data devices can be added to existing thin pools dynamically without needing a system outage. New thin devices can also be created and associated with existing thin pools.

When data devices are added to a thin pool they can be in an enabled or disabled state. In order for the data device to be used for thin extent allocation it needs to be in the enabled state. For it to be removed from the thin pool, it needs to be in a disabled state. A data device can be disabled only if it does not have any thin extents allocated. The following figure depicts the relationships between thin devices and their associated thin pools. There are nine devices associated with thin pool A and three thin devices associated with thin pool B.
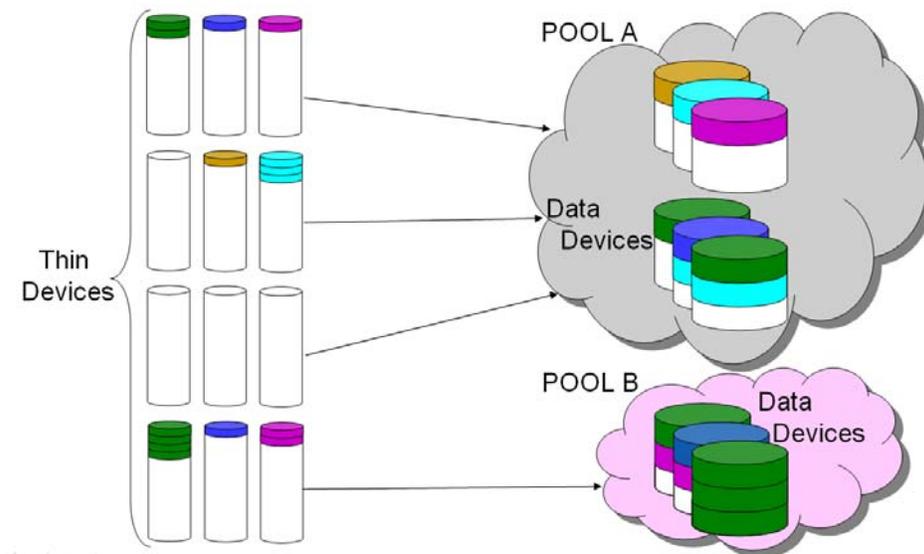
Figure 2. Thin devices and thin pools containing data devices

The way data extents are allocated across the data devices results in a form of striping in the thin pool. The more data devices that are in the thin pool, the wider the striping is and the greater the number of devices that can participate in application I/O. The thin extent size is currently 768 KB in size. This may change in future versions of Enginuity.

The maximum size of a thin device is 240 GB. If a larger size is needed then a metavolume comprised of thin devices can be created. It is recommended that the metavolume be concatenated rather than striped since the thin pool is already striped using thin extents.

Veritas Storage Foundation

Veritas Storage Foundation™ from Symantec provides a complete solution for heterogeneous online storage management. Based on the industry-leading Veritas Volume Manager™ and Veritas File System™, it provides a standard set of integrated tools to centrally manage explosive data growth, maximize storage hardware investments, provide data protection, and adapt to changing business requirements.   Unlike point solutions, Storage Foundation enables IT organizations to manage their storage infrastructure with one tool. With advanced features such as centralized storage management, nondisruptive online configuration and administration, Dynamic Storage Tiering, Dynamic Multipathing, data migration, and local and remote replication, Storage Foundation enables organizations to reduce operational costs and capital expenditures across the data center.

Migrating from fully provisioned storage to thin storage

Current user environments provisioned with traditional (thick or fat) storage are challenged with the task of migrating or deploying into thin storage environments. Users of thin storage are interested in increasing storage utilization rates and reducing storage costs. While there is growing enthusiasm for thin provisioning and its benefits, many customers are particularly interested in realizing these benefits by reclaiming the unused space from existing applications by migrating their thick storage environments into thin environments.

Storage arrays do not track whether a block of data has been written or not. Because of this fact, traditional block-level copy methods cannot pick out the blocks that contain data and only move those blocks from thick storage arrays to thin provisioned arrays. As a result, the traditional migration process completely fills up the new thin LUN, and defeats the purpose of virtually provisioning the storage (as shown in Figure 3).

Some organizations are evaluating "virtual" LUNs. These LUNs are presented by a switch, controller, or an appliance, which in turn mirrors or concatenates LUNs created by more traditional storage arrays. Unfortunately, these approaches must also copy all the blocks during migrations into thin LUNs. So how can one migrate from a traditional storage environment to a thin storage environment without moving every block of data on every LUN?

Solving migration challenges with Veritas SmartMove

The challenges associated with a traditional migration can only be addressed by software that is capable of knowing whether a block is occupied or not. Some solutions work by assuming that free space in the host file system contains all zeroes. In most real life environments, free space actually contains "whatever was written there up until the data was deleted in the file system." To optimally reclaim the unused space in a host file system, one needs to know which blocks support live application data and assume that everything else is free (whether they contain zeroes or not).

Veritas Storage Foundation SmartMove technology uses this knowledge to enable efficient online migrations from thick storage to thin storage, thus reclaiming unused free space.

SmartMove enhances the Veritas Storage Foundation's basic storage migration capability, which has been in the product since its first release in 1990. Tight integration between Veritas Volume Manager and the Veritas File System enables SmartMove to identify the free blocks that do not need to be copied.

SmartMove utilizes this knowledge during online migrations and ensures that only the necessary blocks of data are moved. SmartMove maximizes the value of using virtually provisioned storage systems by allowing applications to automatically and easily reclaim unused space.
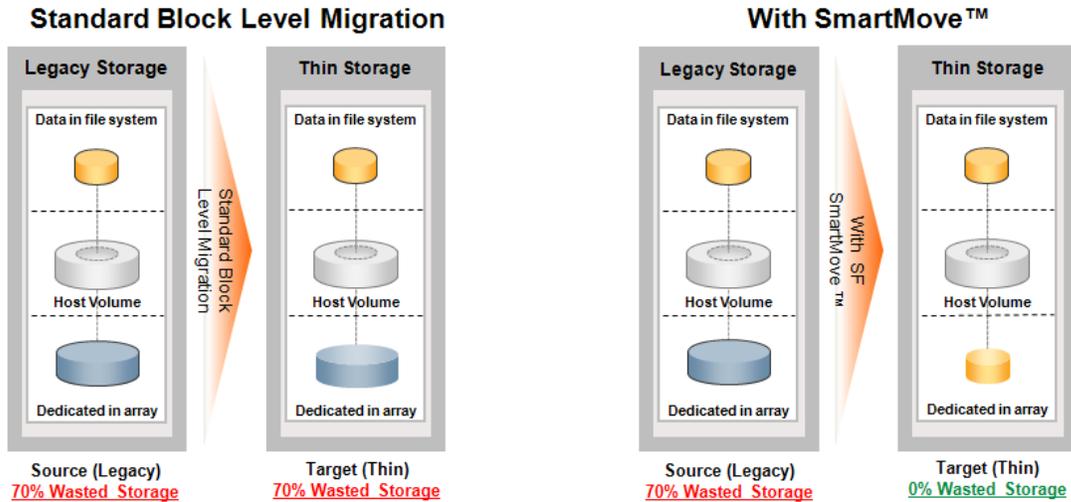
Figure 3. Standard block-level migration vs. SmartMove

Throughout an online migration operation, Veritas Volume Manager (VxVM) requests the free space map from the Veritas File System (VxFS) using SmartMove technology. This is transparent to users and administrators, and the migration steps are identical to those performed in traditional array migrations. When creating the new mirror on the thin LUN, Storage Foundation only copies the occupied blocks. It does not copy the free blocks (Figure 4), regardless of whether they contain zeroes or the remainder of deleted files. By only copying the used space, SmartMove enables faster migrations and ensures that there are no unnecessary storage allocations on the thin LUN.
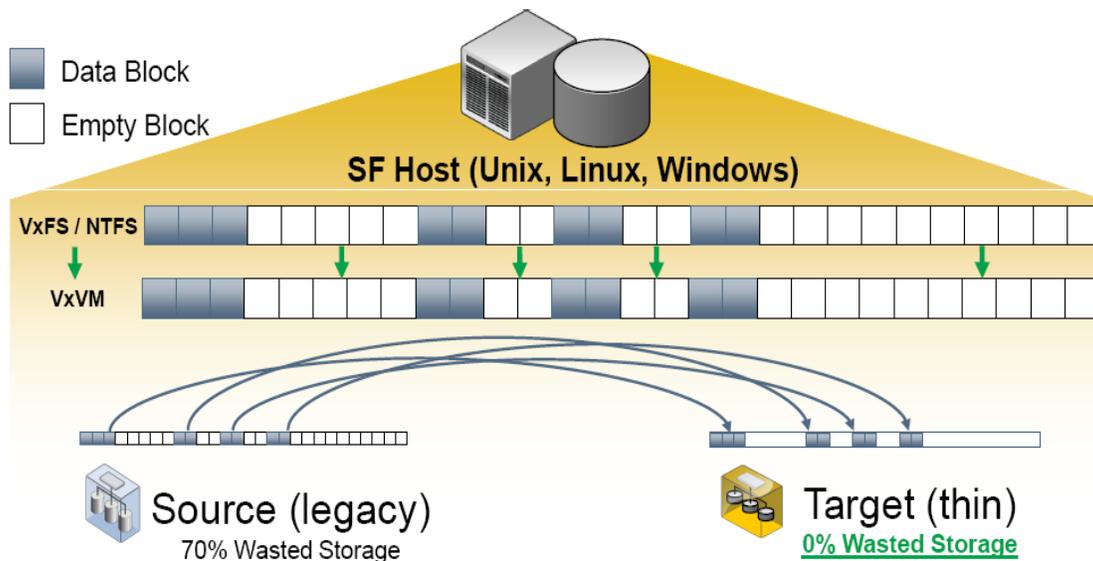
Figure 4. Veritas SmartMove only migrates occupied blocks

On UNIX and Linux, Storage Foundation 5.0 MP3 DMP has the ability to automatically discover the thin LUNs. If any type of block-level copy operation is performed on a device that is known to be thin, SmartMove automatically takes over.

Enabling and using SmartMove with EMC storage

Veritas Storage Foundation 5.0 MP3 automatically uses SmartMove for host volumes that contain EMC virtually provisioned storage devices. This automation is enabled by the EMC-specific Array Support

Library (ASL) that comes with Storage Foundation Dynamic Multipathing (DMP). An ASL contains the set of instructions that allows DMP to properly claim devices during device discovery, allowing DMP to gather device specific attributes and identify the array where the device is located. In the event that the ASL for a given storage array does not discover these device attributes, they can be set manually by a system administrator, thereby allowing the devices to be used for SmartMove operations.

Creating a Symmetrix V-Max environment for thin device migration

Disk devices created from Symmetrix V-Max data devices are managed in the same manner as traditional (thick or fat) devices. Symmetrix V-Max thin devices are configured using either the Symmetrix Command Line Interface (SYMCLI) or Symmetrix Management Console (SMC). Thin pools are created, and then thin devices are added and bound to a pool.

When devices are presented to a host for application initialization, they look identical to other host operating system devices that users are already familiar with.

Creating, adding, and binding devices

Creating thin pools, adding devices to pools, and binding those devices are required steps for creating and configuring a virtually provisioned environment on EMC Symmetrix V-Max.

For this project work, creating devices, binding them to a thin pool, and masking/mapping the devices to a host was performed using the following command line interface command syntax:

1. **symconfigure -sid 186 -cmd "create pool SmartPool type=thin;" commit**

    a. A thin pool named SmartPool is created.

    > Note: Thin pools can be created without adding data devices so that they can be populated at a later time.

2. **symconfigure -sid 186 -cmd "create dev count=3, size=3 GB, emulation=fba, in pool=SmartPool member_state=enable config=2-Way-Mir, attribute=datadev;" commit**

    a. This command creates three data devices from standard RAID 1 (2-Way-Mir) devices. The devices must have an emulation type of FBA for open systems hosts. The sizing designation is at the user's discretion. For this example, the pool is created with three devices that are each 3 GB.

    > Note: The output listing of the above command includes the new device IDs that will be needed to create the thin devices.

3. **symconfigure -sid 186 -cmd "create dev count=3, size=3 GB, emulation=fba, config=tdev binding to pool=SmartPool;" commit**

    a. Create three thin devices and bind them to the SmartPool pool.

    > Note: The output of the command displays the thin device IDs 3C32:3C34.

4. **symconfigure –sid 186 –cmd "map dev 3C32:3C34 to dir 10A:1 starting lun=32;" commit**

    symconfigure –sid 186 –cmd "map dev 3C32:3C34 to dir 7A:0 starting lun=32;" commit

    a. Map the thin devices to the two front-end adapter (FA) ports.

5. **symmask –sid 186 –wwn 10000000c9563968 add devs 3C32:3C34 –dir 10A –p 1**

    symmask –sid 186 –wwn 10000000c9563969 add devs 3C32:3C34 –dir 7A –p 0

a. Mask the devices to the host

Note: It will be necessary to rescan the host buses or reboot, in order to make the devices visible to the host.

6. **symcfg discover**

a. After rescan or reboot, issue this command to rediscover available devices.

7. **sympd list**

```
LicoC035# sympd list -sid 186

Symmetrix ID: 000190102186

     Device Name        Directors           Device
------------------------ ------------ --------------------------------
                                                Cap
Physical          Sym SA :P DA :IT Config     Attribute  Sts  (MB)
------------------------ ------------ --------------------------------
c4t50060482D52EAAA9d1* 0760 10A:1 NA:NA  TDEV        N/Grp'd   RW   4314
c4t50060482D52EAAA9d2* 0761 10A:1 NA:NA  TDEV        N/Grp'd   RW   4314
 …
c4t50060482D52EAAA9d4* 0892 10A:1 01A:C4 RAID-5      N/Grp'd   RW   4315
c4t50060482D52EAAA9d4* 0893 10A:1 01B:CB RAID-5      N/Grp'd   RW   4315
 …
c4t50060482D52EAAA9d6* 3C32 10A:1 NA:NA  TDEV        N/Grp'd   RW   3072
c4t50060482D52EAAA9d6* 3C33 10A:1 NA:NA  TDEV        N/Grp'd   RW   3072
 …
c5t50060482D52EAA86d1* 09DC 07A:0 02C:D6 2-Way Mir   N/Grp'd   RW   4314
c5t50060482D52EAA86d1* 09DD 07A:0 15B:C5 2-Way Mir   N/Grp'd   RW   4314
…
rdmp/emc0_076as2      076A 07A:0 NA:NA  TDEV        N/Grp'd   RW   4314
rdmp/emc0_076bs2      076B 07A:0 NA:NA  TDEV        N/Grp'd   RW   4314
…
```

Figure 5. Symmetrix V-Max devices available to the host

Figure 5 displays a Symmetrix V-Max physical device listing showing available LUNs for the host. The devices labeled "TDEV" are the thin devices, "2-Way Mir" are standard (thick) RAID 1 devices, and there are standard RAID 5 devices, as well as the Veritas SF devices. All devices were configured on two separate director (DA) paths: 10A:1 and 7A:0. This allows for dynamic multipathing and I/O load balancing using either EMC PowerPath® or Symantec DMP.

Note: Not all available devices are shown in the output listing.

Discovering thin devices using Storage Foundation

Veritas Storage Foundation discovers thin devices automatically on the EMC Symmetrix DMX (5773 or later) or V-Max hardware. Veritas Storage Foundation software will list thin devices that have been recognized and initialized on the host as "thin."

Using Storage Foundation enclosure based naming (EBN), the EMC devices are named with an "emc0_" prefix. The user can associate the enclosure based device names to the EMC device ID by using the "vxdisk list" and "sympd list" output listings, and then compare the device IDs.

```
LicoC035#   vxdisk -o thin list

DEVICE        SIZE(mb)   PHYS_ALLOC(mb) GROUP      TYPE

emc0_076a     4314       0          -          thin
emc0_076b     4314       0          -          thin
emc0_076c     4314       0          -          thin
emc0_076d     4314       0          -          thin
emc0_076e     4314       0          -          thin
...

LicoC035# sympd list

Symmetrix ID: 000190102186

     Device Name        Directors        Device
-------------------------- ------------- -------------------------------------
                                            Cap
Physical        Sym_SA :P DA :IT Config     Attribute   Sts  (MB)
-------------------------- ------------- -------------------------------------


c4t50060482D52EAAA9d1* 076A 10A:1 NA:NA TDEV      N/Grp'd   RW   4314
c4t50060482D52EAAA9d1* 076B 10A:1 NA:NA TDEV      N/Grp'd   RW   4314
c4t50060482D52EAAA9d1* 076C 10A:1 NA:NA TDEV      N/Grp'd   RW   4314
c4t50060482D52EAAA9d1* 076D 10A:1 NA:NA TDEV      N/Grp'd   RW   4314
c4t50060482D52EAAA9d1* 076E 10A:1 NA:NA TDEV      N/Grp'd   RW   4314
 ...
```

Figure 6. Comparing EBN to the EMC name (highlighted)

All devices that have been presented to the host and recognized by Storage Foundation can be viewed in the same listing as follows.

vxdisk list

```
LicoC035# vxdisk list

DEVICE          TYPE           DISK      GROUP       STATUS

emc0_09df       auto:cdsdisk   -         -           online
emc0_09e0       auto:cdsdisk   -         -           online
emc0_09e1       auto:cdsdisk   -         -           online
. . .
emc0_076a       auto:cdsdisk   -         -           online thin
emc0_076b       auto:cdsdisk   -         -           online thin
emc0_076c       auto:cdsdisk   -         -           online thin
emc0_0764       auto:cdsdisk   -         -           online thin
emc0_0765       auto:cdsdisk   -         -           online thin
emc0_0766       auto:cdsdisk   -         -           online thin
emc0_0767       auto:cdsdisk   -         -           online thin
emc0_0768       auto:cdsdisk   -         -           online thin
. . .
emc0_0892       auto:cdsdisk   -         -           online
emc0_0893       auto:cdsdisk   -         -           online
. . .
```

Figure 7. Veritas Storage Foundation disk listing output

By default, SmartMove applies to all copy operations performed on VxVM volumes that contain thin devices, and that support a mounted VxFS file system. On UNIX and Linux, SmartMove can be controlled by a host wide setting in the /etc/default/vxsf file: usefsSmartMove. This tunable can be set to none, all, or thinonly:

usefsSmartMove=none: SmartMove is disabled on the host.

usefsSmartMove=all: SmartMove is enabled for all copy operations on the host.

usefsSmartMove=thinonly: SmartMove is enabled for volumes that contain thin devices.

The default Storage Foundation behavior is equivalent to having the usefsSmartMove tunable set to 'thinonly'.

Migrating thick to thin storage

There are multiple ways to perform online storage migrations with Veritas Volume Manager. Best practice is to use the vxassist command. The procedure to perform a thick-to-thin migration is strictly identical to the procedure that would be used to perform any other storage migration using Veritas Volume Manager because SmartMove automatically kicks in to ensure optimized migrations to thin devices.

Best practices would dictate that the following steps be used to perform a thick-to-thin device migration.

1. **Create a VxVM disk group:**

   vxdg init DataDG emc0_0896 emc0_0897 emc0_0898 emc0_3c32 emc0_3c33 emc0_3c34

   a. A device group is initialized for ease of use and portability. The device group named DataDG contains six devices. The first three — emc0_0896, emc0_0897, and emc0_0898 — are thick devices. The last three — emc0_3c32, emc0_03c33, and emc0_3c34 — are thin devices.

2. **Create a VxVM volume:**

   vxassist -g DataDG make datavol 2500m emc0_0896

   a. This creates a volume named datavol on the thick device emc0_0896. The thick device emc0_0896 is of size 4 GB. The size of the volume "datavol" that is created on the thick device will be approximately 2.5 GB.

3. **Check the host configuration:**

   vxprint –ht –g DataDG

   a. This command shows the current configuration of the disk group DataDG.

```
dm emc0_3c32      emc0_3c32      auto      65536     6222208   -
dm emc0_3c33      emc0_3c33      auto      65536     6222208   -
dm emc0_3c34      emc0_3c34      auto      65536     6222208   -
dm emc0_0896      emc0_0896      auto      65536     8761088   -
dm emc0_0897      emc0_0897      auto      65536     8761088   -
dm emc0_0898      emc0_0898      auto      65536     8761088   -


v  datavol        -              ENABLED   ACTIVE    6291456   SELECT    -          fsgen
pl datavol-01     datavol        ENABLED   ACTIVE    6291456   CONCAT    -          RW
sd emc0_0896-01   datavol-01     emc0_0896 0         6291456   0         emc0_0896 ENA
```

Figure 8. The last line in the command output shows the volume "datavol" is using thick device emc0_0896

4. **Make a VxFS file system on the datavol volume:**

   mkfs –F vxfs /dev/vx/rdsk/DataDG/datavol

   a. Makes a file system on the volume datavol.

5. **Create a mount point for the VxFS file system:**

   mkdir /data/smartfs

   a. Makes a directory on the host to mount the file system.

6. **Mount the VxFS file system:**

   mount -F vxfs /dev/vx/dsk/DataDG/datavol /data/smartfs

   a. Mounts the file system, as confirmed in Figure 9:

```
LicoC035# df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t2d0s0    58247486 22690918 34974094    40%    /
/devices                   0        0        0     0%    /devices
ctfs                       0        0        0     0%    /system/contract
proc                       0        0        0     0%    /proc
mnttab                     0        0        0     0%    /etc/mnttab
swap                 10028356      956 10027400     1%    /etc/svc/volatile
objfs                      0        0        0     0%    /system/object
/usr/lib/libc/libc_hwcap2.so.1
                     58247486 22690918 34974094    40%    /lib/libc.so.1
fd                         0        0        0     0%    /dev/fd
swap                 10027440       40 10027400     1%    /tmp
swap                 10027432       32 10027400     1%    /var/run
swap                 10027400        0 10027400     0%    /dev/vx/dmp
swap                 10027400        0 10027400     0%    /dev/vx/rdmp
/dev/dsk/c0t3d0s0    60509964 27211602 32693263    46%    /orig_s0
/dev/dsk/c0t3d0s7     8029464     7985 7941185     1%    /orig_s7
/dev/dsk/c0t2d0s7     8068883     8025 7980170     1%    /export/home
/dev/vx/dsk/DataDG/datavol
                      2560000    18127 2383013     1%    /data/smartfs
```

Figure 9. The smartfs file system is empty at this time (used capacity = 1%)

7. **Populate the file system by copying files into it, or creating an application on the file system, etc.**

For the purpose of this exercise, a 1.4 GB tar file was copied: cp data.tar /data/smartfs

```
/dev/dsk/c0t3d0s0    60509964 27211602 32693263    46%    /orig_s0
/dev/dsk/c0t3d0s7     8029464     7985 7941185     1%    /orig_s7
/dev/dsk/c0t2d0s7     8068883     8025 7980170     1%    /export/home
/dev/vx/dsk/DataDG/datavol
                      2560000  1400626 1086920    57%    /data/smartfs
```

Figure 10. Display the file system size after the copy

The file system was created of size 2.5 GB, therefore by copying a 1.4 GB file, we see that 57% of the file system is now consumed. The file system "/data/smartfs" is of size 2.5 GB on a thick device of 4 GB. After copying a file of 1.4 GB on to this file system, only 1.4 GB of space is being used out of the 4 GB space allocated to the thick volume emc0_0896, resulting in 2.6 GB of allocated but unused storage space.

8. **Before the migration, check utilization of the thin pool in the EMC array:**

   symcfg show -sid 186 -pool SmartPool -thin -details -all –mb

   a. The output of this command shows the amount of physical disk space used in the SmartPool. Prior to the migration, the thin volume 3C32 is empty and just 1 MB is used for metadata.

```
LicoC035# symcfg show -sid 186 -pool SmartPool -thin -details -all -mb

Symmetrix ID: 000190102186

Thin Devices(3):
  -------------------------------------------------------------------
                 Pool          Pool              Pool
   Sym      Total Subs       Allocated          Written
   Dev       MBs  (%)     MBs  (%)MBs         (%) Status
  -------------------------------------------------------------------
   3C32      3072  33       1    0     -       -   Bound
   3C33      3072  33       1    0     -       -   Bound
   3C34      3072  33       1    0     -       -   Bound
          ---------- ----- ---------- ----- ---------- -----
   MBs      9217  100       3    0     -       -
```

Figure 11. Pool Allocated (0%) shows that the pool named SmartPool is empty

9. **Perform the thick-to-thin migration using Veritas Volume Manager:**

   vxassist -g DataDG move datavol !emc0_0896 emc0_3c32

   a. This Veritas Volume Manager command performs the migration operation by copying the contents of the volume "datavol" from thick device emc0_0896 to thin device emc0_3c32. The purpose of this operation is to reclaim the 2.6GB of allocated but unused space on the thick device emc0_0896 by migrating only the data blocks used by the file system "/data/smartfs" to the thin device emc0_3c32. Note that the migration of the data blocks is an online operation (the file system is mounted and active).

10. **After the migration, verify that datavol is now supported by thin device emc0_3c32:**

   vxprint –ht –g DataDG

   a. Execute this command again to see the current configuration of the volume "datavol" in the disk group DataDG. Note that the datavol is using the new thin device emc0_3c32.

```
v  datavol        -              ENABLED  ACTIVE   5120000  SELECT   -            fsgen
pl datavol-01     datavol        ENABLED  ACTIVE   5120000  CONCAT   -            RW
sd emc0_3c32-01   datavol-01     emc0_3c32 0       5120000  0        emc0_3c32 ENA
```

Figure 12. vxprint output after migration

11. **Finally, check utilization of the thin pool in the EMC array:**

   symcfg show -sid 186 -pool SmartPool -thin -details -all –mb

   a. The output of this command shows the amount of physical disk space allocated to the thin devices in SmartPool after migration.

```
LicoC035# symcfg show -sid 186 -pool SmartPool -thin -details -all -mb

Thin Devices(3):
     -----------------------------------------------------------------
                   Pool        Pool        Pool
     Sym      Total  Subs    Allocated    Written
     Dev        MBs  (%)      MBs  (%)     MBs  (%)  Status
     -----------------------------------------------------------------
     3C32       3072  33     1377   15      -    -   Bound
     3C33       3072  33        1    0      -    -   Bound
     3C34       3072  33        1    0      -    -   Bound
              ---------- ----- ---------- ----- ---------- -----
     MBs        9217  100    1379   15      -    -
```

Figure 13. symcli output listing of the thin pool after the SmartMove migration

Figure 13 shows that only 1,377 MB, or approximately 1.4 GB, of space is allocated to thin device 3C32. This is exactly equal to the amount of used space in the smartfs file system that we migrated (that file system contains a single 1.4 GB tar file).

To summarize, Storage Foundation SmartMove allowed us to perform an online migration of a 2.5 GB file system with 1.4 GB of used space from a 4 GB thick device (emc0_0896) to a 3 GB thin device (emc0_03c32) with only 1.4 GB of physical space allocated to it. This migration allowed us to automatically and transparently reclaim 2.6 GB (4 GB of emc0_0896– 1.4 GB of emc0-3c32) of physical storage.

Optimal allocation of thin storage

Using Veritas Storage Foundation SmartMove, users can reclaim stranded thick storage by migrating online only the used data blocks to a thin storage. Thin storage starts off as thin, but over a period of time becomes not so thin. This happens due to the fact that thin storage works on the principle "write once, allocate forever." So the need for optimal sizing of thin storage devices is clear.

It is possible to use SmartMove as a method of grooming thin storage by performing thin-to-thin migrations. Creating spare devices and using EMC's SYMCLI or SMC allows users to track and manage pool consumption. This solution can be used with SmartMove support of a thin-to-thin migration.

Here, building on the previous section and examples, one thin device will be migrated to another. This example shows how space is optimized after the migration completes.

1. **Copy a 500 MB file on the file system /data/smartfs.**

     a. Prior to copying a 500 MB file, /data/smartfs is using a total of 1.4 GB on the file system. The thin device emc0_3c32 on which the file system is created allocated exactly 1.4 GB of storage. After the copy operation the file system will be using approximately 1.9 GB of space (Figure 14) and SmartPool will allocate an additional 500 MB to the thin device as seen in Figure 15.

```
/dev/dsk/c0t3d0s0     60509964 27211602 32693263    46%    /orig_s0
/dev/dsk/c0t3d0s7      8029464     7985 7941185     1%    /orig_s7
/dev/dsk/c0t2d0s7      8068883     8025 7980170     1%    /export/home
/dev/vx/dsk/DataDG/datavol
                      2560000 1916079   603683    77%    /data/smartfs
```

Figure 14. "df –k" output showing the used space of 1.9 GB on the file system /data/smartfs

```
LicoC035# symcfg show -sid 186 -pool SmartPool -thin -details -all -mb

Symmetrix ID: 000190102186


Thin Devices(3):
 {
  ---------------------------------------------------------------------
            Pool      Pool         Pool
  Sym      Total Subs    Allocated     Written
  Dev       MBs  (%)    MBs  (%)     MBs  (%) Status
  ---------------------------------------------------------------------
  3C32      3072  33     1881  20    -    - Bound
  3C33      3072  33      1   0    -    - Bound
  3C34      3072  33      1   0    -    - Bound
       ---------- ----- ---------- ----- ---------- -----
  MBs       9217 100     1883  20    -    -
```

Figure 15. symcli output showing space allocation for the thin device 3c32 (1881 MB or 1.9 GB)

Note that thin device 3C33 is totally unused, with 0% allocated. This device can be used as the target of our thin-to-thin migration.

2. **Delete the data.tar file from the /data/smartfs file system**.

    a. This delete will free up 1.4 GB of space from the file system, resulting in 500 MB used space on the file system in Figure 16. Even though the file system released the 1.4 GB of space because of the file deletion, this space is not freed up on the thin device emc0_3c32.

```
/dev/dsk/c0t3d0s0     60509964 27211602 32693263    46%    /orig_s0
/dev/dsk/c0t3d0s7      8029464     7985 7941185     1%    /orig_s7
/dev/dsk/c0t2d0s7      8068883     8025 7980170     1%    /export/home
/dev/vx/dsk/DataDG/datavol
                      2560000  530128 1903012    22%    /data/smartfs
```

Figure 16. "df –k" output showing 500 MB of used space on the /data/smartfs file system

3. **Perform the thin-to-thin migration using Veritas Volume Manager:**

    vxassist -g DataDG move datavol !emc0_3c32 emc0_3c33

a. Perform a move (or copy) operation by migrating the contents of the volume from thin device emc0_3c32 to thin device emc0_3c33 in disk group DataDG in Figure 17. The purpose of this operation is to reclaim the 1.4 GB of allocated but unused space on the thin device emc0_3c32 by migrating only the data blocks used by the file system "/data/smartfs" to the thin device emc0_3c33.

Note that the migration of the data blocks is an online operation (the file system is mounted and active).

```
LicoC035# vxdisk -g DataDG list
DEVICE        TYPE          DISK         GROUP       STATUS
emc0_3c32     auto:cdsdisk  emc0_3c32    DataDG      online thin
emc0_3c33     auto:cdsdisk  emc0_3c33    DataDG      online thin
emc0_3c34     auto:cdsdisk  emc0_3c34    DataDG      online thin
emc0_0896     auto:cdsdisk  emc0_0896    DataDG      online
emc0_0897     auto:cdsdisk  emc0_0897    DataDG      online
emc0_0898     auto:cdsdisk  emc0_0898    DataDG      online
```

Figure 17. vxdisk list output showing the thin device emc0_3c33 as part of the DataDG disk group

4. **After the migration, verify that datavol is now supported by thin device emc0_3c33:**

   vxprint –ht –g DataDG

   a. Execute this command again to see the current configuration of the volume "datavol" in the disk group DataDG. Note that datavol is using the new thin device emc0_3c33.

```
v  datavol        -               ENABLED  ACTIVE   5120000  SELECT   -          fsgen
pl datavol-01     datavol         ENABLED  ACTIVE   5120000  CONCAT   -          RW
sd emc0_3c33-01   datavol-01      emc0_3c33 0        5120000  0        emc0_3c33 ENA
```

Figure 18. vxprint output after migration

5. **Finally, check the allocated space on device 3c33 and verify that space was reclaimed:**

   symcfg show -sid 186 -pool SmartPool -thin -details -all –mb

   a. The output of this command shows the amount of physical disk space allocated to the thin devices in SmartPool after migration.

```
LicoC035# symcfg show -sid 186 -pool SmartPool -thin -details -all -mb

Symmetrix ID: 000190102186

Thin Devices(3):
  {
    -------------------------------------------------------------------
                Pool        Pool        Pool
    Sym     Total  Subs   Allocated    Written
    Dev      MBs   (%)    MBs  (%) MBs  (%) Status
    -------------------------------------------------------------------
    3C32     3072   33    1881  20   -    -  Bound
    3C33     3072   33     527   6   -    -  Bound
    3C34     3072   33       1   0   -    -  Bound
    ---------- ----- ---------- ----- ---------- -----
    MBs      9217  100    2409  26   -    -
```

Figure 19. symcli output listing of the thin pool after the SmartMove migration

Figure 19 shows that 527 MB of space was allocated to the thin device 3C33 specified during the SmartMove migration, confirming that the operation automatically right-sized the thin storage to the minimum required to support written data in the file system.

By using this method of thin-to-thin migration, only the allocated and used space in thin device emc_3c32 is migrated to thin device emc_3c33. The allocated but unused storage in thin device emc_3c32 can be reclaimed by releasing the space to the storage pool.

Conclusion

Virtual Provisioning technology enables organizations to improve speed and ease of use, enhance performance, and increase capacity utilization for certain applications and workloads. EMC Symmetrix Virtual Provisioning integrates with existing device management, replication, and other management tools, enabling customers to easily build Virtual Provisioning into their existing storage management processes. Veritas Storage Foundation SmartMove by Symantec enables organizations to fully optimize EMC Symmetrix Virtual Provisioning by migrating online to thin storage and automatically reclaiming all unused space. The integration of EMC Virtual Provisioning and Veritas Storage Foundation SmartMove technologies closes a gap and allows users to easily migrate their thick storage environments to thin provisioning without application downtime so they can take advantage of the operational benefits of EMC Virtual Provisioning.

References

- *Veritas File System Administrator's Guide 5.0 MP3 – Solaris*
- *Veritas Volume Manager Administrator's Guide 5.0 MP3 – Solaris*
- *Best Practices for Fast, Simple Capacity Allocation with EMC Symmetrix Virtual Provisioning Technical Note*