

Dell EMC SC Series: Best Practices with VMware vSphere

Abstract

This document provides best practices for integrating VMware® vSphere® 5.x-7.x hosts with Dell EMC™ SC Series storage.

May 2021

Revisions

Date	Description
July 2016	Initial release: Combined vSphere 5.x and 6.x best practice documents, added SCOS 7.1 updates
September 2016	Minor revisions and corrections
October 2016	Changed Disk.AutoremoveOnPDL to reflect current VMware guidance
January 2017	Updated for vSphere 6.5 changes; added appendix D summarizing all host settings
February 2017	Updated Linux guest disk timeout recommendations in section 4.7.2
April 2017	Updated iSCSI login timeout recommendation
July 2017	Updated SAS driver info in 4.2.3, Added auto UNMAP requirements in 16.3.5
April 2018	Updated to provide vSphere 6.7 guidance
October 2018	Minor revisions and corrections
December 2018	Added SAS front-end lsi_msgpt3 module parameter recommendation in 4.2.3
March 2019	Modified SATP claim rules in section 6.9.1 and appendix D
July 2019	Minor revisions and corrections (VMFS3.EnableBlockDelete=1)
September 2019	Claim rule syntax corrections
April 2020	vSphere 7.0 additions
December 2020	Minor clarifications
January 2021	Esxcli command syntax change for DelayedAck in appendix D.1
May 2021	Updated section 4.2.3 with additional KB articles for SAS FE connectivity

Acknowledgments

Author: Darin Schmitz

The information in this publication is provided "as is." Dell Inc. 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 software described in this publication requires an applicable software license.

This document may contain certain words that are not consistent with Dell's current language guidelines. Dell plans to update the document over subsequent future releases to revise these words accordingly.

This document may contain language from third party content that is not under Dell's control and is not consistent with Dell's current guidelines for Dell's own content. When such third party content is updated by the relevant third parties, this document will be revised accordingly.

Copyright © 2016–2021 Dell Inc. or its subsidiaries. All Rights Reserved. Dell Technologies, Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners. [5/24/2021] [Best Practices] [2060-M-BP-V]

Table of contents

Revisions.....	2
Acknowledgments.....	2
Table of contents	3
1 Introduction.....	6
1.1 Audience.....	6
1.2 Prerequisites.....	6
2 Fibre Channel switch zoning	7
2.1 Single initiator multiple target zoning.....	7
2.2 WWN zoning.....	7
2.3 Port zoning.....	7
2.4 Virtual ports.....	8
3 Host initiator settings	9
4 Modifying queue depth and timeouts	10
4.1 Host bus adapter queue depth	10
4.2 Storage driver queue depth and timeouts	10
4.3 Adjusting settings for permanent device loss conditions.....	13
4.4 Modifying the VMFS queue depth for virtual machines (DSNRO)	13
4.5 Adaptive queue depth.....	16
4.6 Modifying the guest operating system queue depth.....	16
4.7 Setting operating system disk timeouts.....	17
5 Guest virtual SCSI adapter selection	19
5.1 BusLogic Parallel.....	19
5.2 LSI Logic Parallel.....	19
5.3 LSI Logic SAS.....	19
5.4 VMware Paravirtual	19
6 Mapping volumes to an ESXi server	20
6.1 Basic volume mapping concepts	20
6.2 Basic SC Series volume mappings	20
6.3 Multipathed volume concepts	21
6.4 Multipathed SC Series volumes	22
6.5 Configuring the VMware iSCSI software initiator for a single path.....	24
6.6 Configuring the VMware iSCSI software initiator for multipathing.....	25
6.7 iSCSI port multi-VLAN configuration recommendations.....	27
6.8 Configuring the FCoE software initiator for multipathing.....	28

6.9	VMware multipathing policies	28
6.10	Multipathing using a fixed path selection policy	32
6.11	Multipathing using a round robin path selection policy	33
6.12	Asymmetric logical unit access (ALUA) for front-end SAS	33
6.13	Unmapping volumes from an ESXi host	34
6.14	Mapping volumes from multiple arrays	35
6.15	Multipathing resources	35
7	Boot from SAN	36
7.1	Configuring boot from SAN	36
8	Volume creation and sizing	38
8.1	Volume sizing and the 64 TB limit	38
8.2	Virtual machines per datastore	38
8.3	VMFS partition alignment	39
8.4	VMFS file systems and block sizes	41
9	Volume mapping layout	42
9.1	Multiple virtual machines per volume	42
9.2	One virtual machine per volume	45
10	Raw device mapping (RDM)	46
11	Data Progression and storage profile selection	47
11.1	On-Demand Data Progression	48
11.2	Data reduction (compression and deduplication)	49
12	Thin provisioning and virtual disks	51
12.1	Virtual disk formats	51
12.2	Thin provisioning relationship	52
12.3	SC Series thin write functionality	52
12.4	SC Series thin provisioning or VMware thin provisioning	52
12.5	Windows free space recovery	52
12.6	Affinity Manager 2.0	53
13	Extending VMware volumes	54
13.1	Increasing the size of VMFS datastores	54
13.2	Increasing the size of a virtual machine disk (VMDK) file	55
13.3	Increasing the size of a raw device mapping (RDM)	55
14	Snapshots (replays) and virtual machine backups	56
14.1	Backing up virtual machines	56
14.2	Recovering virtual machine data from a snapshot	57
15	Replication and remote recovery	60

- 15.1 Synchronous replication60
- 15.2 Asynchronous replication60
- 15.3 Replication considerations with standard replications.....61
- 15.4 Replication considerations with Live Volumes61
- 15.5 Replication tips and tricks.....62
- 15.6 Virtual machine recovery at a DR site63
- 16 VMware storage features64
 - 16.1 Storage I/O Controls (SIOC).....64
 - 16.2 Storage distributed resource scheduler (SDRS)66
 - 16.3 vStorage APIs for array integration (VAAI).....67
 - 16.4 vStorage APIs for Storage Awareness (VASA).....69
 - 16.5 Virtual Volumes (vVols)70
- A Determining the appropriate queue depth for an ESXi host71
 - A.1 Fibre Channel71
 - A.2 iSCSI.....72
 - A.3 Using esxtop to monitor queue depth.....72
- B Deploying vSphere client plug-ins74
 - B.1 Dell Storage vSphere Web Client plug-in.....74
- C Configuring Dell Storage Manager VMware integrations.....75
- D Host and cluster settings76
 - D.1 Recommended settings.....76
 - D.2 Optional settings.....77
- E Additional resources.....78
 - E.1 Technical support and resources78
 - E.2 VMware support.....78

1 Introduction

This document provides configuration examples, tips, recommended settings, and other storage guidelines for integrating VMware® vSphere® hosts with the Dell EMC™ SC Series storage. It also answers many frequently asked questions about how VMware interacts with SC Series features like Dynamic Capacity (thin provisioning), Data Progression (automated tiering), and Remote Instant Replay (replication).

1.1 Audience

This technical document is intended for storage and server administrators, and other information technology professionals interested in learning more about how VMware vSphere integrates with SC Series storage.

1.2 Prerequisites

Understanding the material in this document requires formal training or advanced working knowledge of the following:

- Installation and configuration of VMware vSphere 5.x/6.x/7.x
- Configuration and operation of the SC Series
- Operation of Dell EMC Storage Manager (DSM)/Enterprise Manager (EM) software
- Using operating systems such as Microsoft® Windows® or Linux®

For important information about configuring VMware ESXi® hosts to use the SAN, refer to the appropriate *vSphere Storage Guide*: [VMware vSphere Documentation](#).

Note: This document provides general recommendations that may not be applicable to all configurations or needs.

2 Fibre Channel switch zoning

Zoning Fibre Channel switches for an ESXi host is like zoning any other server connected to the SC Series array. The fundamental points are explained in this section.

2.1 Single initiator multiple target zoning

Each Fibre Channel zone created should have a single initiator (HBA port) and multiple targets (SC Series front-end ports). Each HBA port requires its own Fibre Channel zone that contains itself and the SC Series front-end ports. Independent zones should be created for each HBA installed in the host.

2.2 WWN zoning

When zoning by WWN, the zone only needs to contain the host HBA port and the SC Series front-end ports. In legacy port mode, it is not necessary to include the SC Series front-end reserve ports because they are not used for volume mappings. In virtual port mode, the HBA port should be zoned with the virtual port WWNs. For example, if the host has two HBAs connected to two disjointed fabrics, the Fibre Channel zones would look like the configuration shown in Table 1.

Table 1 Example of zoning by WWN

Name	WWN	Description
ESX1-HBA1 (zone created in fabric 1)	2100001B32017114	ESX1 HBA port 1
	5000D31000036001	Controller1 front-end primary plugged into fabric 1
	5000D31000036009	Controller2 front-end primary plugged into fabric 1
ESX1-HBA2 (zone created in fabric 2)	210000E08B930AA6	ESX1 HBA port 2
	5000D31000036002	Controller1 front-end primary plugged into fabric 2
	5000D3100003600A	Controller2 front-end primary plugged into fabric 2

2.3 Port zoning

Caution: Due to the supportability of port zoning, WWN zoning is preferred over port zoning.

Port zoning is creating zones by including the physical ports instead of the WWN. Although this method has security advantages, it creates supportability challenges in the environment.

2.4 Virtual ports

If the SC Series array is configured to use virtual port mode, include all the front-end virtual ports within each fault domain in the zone with each ESXi initiator. See Figure 1.

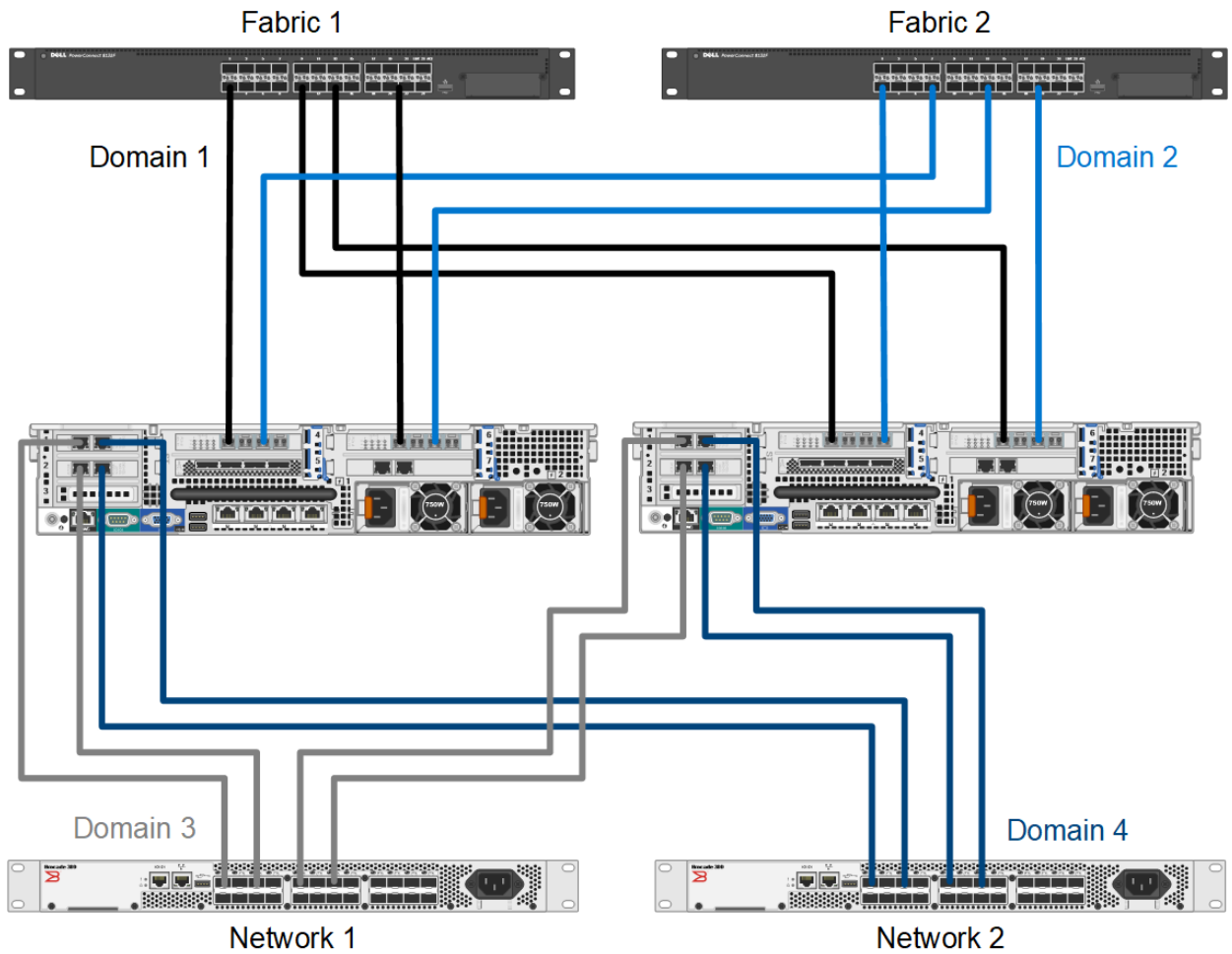


Figure 1 Virtual port domains, Fibre Channel (FC) and iSCSI

3 Host initiator settings

Ensure the initiator settings are configured in the ESXi host according to *Appendix A: Required Adapter and Server OS Settings* in the latest [Dell Storage Compatibility Matrix](#).

The current recommended settings are:

- QLogic® Fibre Channel card BIOS settings:
 - Set connection options to 1 for point to point only
 - Set login retry count to 60 attempts
 - Set port down retry to 60 attempts
 - Set link down timeout to 30 seconds
 - Set the Execution Throttle to 255 (if available). The ESXi VMkernel driver module and DSNRO control the queue depth. According to the QLogic support article, [HBA Execution Throttle And Queue Depth In A VMware Environment](#), the Execution Throttle variable is not used by the QLogic driver.
- Emulex Fibre Channel card BIOS settings:
 - Set the Node Time Out field, `lpfc_devloss_tmo` (formerly `nodev_tmo`) to 60 seconds
 - Set topology to 2 for Auto Topology (point to point first)
 - Set queue depth to 255. A queue depth of 255 allows the ESXi VMkernel driver module and DSNRO to more conveniently control the queue depth.
- QLogic iSCSI HBAs: The ARP redirect must be enabled for controller failover to work properly with iSCSI HBAs that support full hardware offload. Here is an example script for enabling ARP redirect:

```
esxcli iscsi physicalnetworkportal param set --option ArpRedirect -v=1 -A
vmhba4
```

Note: This command will not work with dependent hardware iSCSI adapters. This command replaces the former `esxcfg-hwiscsi` command found in [VMware KB Article 1010309](#). See the vSphere 5 Documentation Center article, [Reference to Replacements for Service Console Commands](#), for an explanation of the new syntax.

- iSCSI initiator settings for delayed ACK: During periods of high network congestion in some environments, iSCSI transfer latency may exceed acceptable levels. VMware recommends disabling delayed ACK using the steps described in the article, [ESX/ESXi hosts might experience read or write performance issues with certain storage arrays](#), in the VMware Knowledge Base.
- SAS HBA card BIOS settings: For Dell Storage SCv2000 Series arrays configured with SAS front-end ports, factory default settings within the card BIOS are recommended for SAS ESXi host HBAs.

4 Modifying queue depth and timeouts

Queue depth is defined as the number of disk transactions that can be in flight between an initiator and a target. The initiator is an ESXi host HBA port or iSCSI initiator, and the target is the SC Series front-end port.

Since any given target can have multiple initiators sending it data, the initiator queue depth is used to throttle the number of transactions. Throttling transactions keeps the target from becoming flooded with I/O. When this flooding happens, the transactions start to pile up, causing higher latencies and degraded performance. While increasing the queue depth can sometimes increase performance, if it is set too high, there is an increased risk of overdriving the storage array.

When data travels between the application and the storage array, there are several places where the queue depth can be set to throttle the number of concurrent disk transactions. The most common places where queue depth can be modified are listed in Table 2.

Table 2 Areas for setting queue depth

Area	Setting
The application itself	Default=dependent on application
The virtual SCSI card driver in the guest	Default=32
The virtual machine file system (VMFS) layer (DSNRO)	Default=32
The HBA VMkernel Module driver	Default=64
The HBA BIOS	Default=varies

The remainder of this section explains how to set the queue depth in each layer.

Caution: The appropriate queue depth for a host may vary due to several factors. As a best practice, only increase or decrease the queue depth if necessary. See appendix A for determining the proper queue depth.

4.1 Host bus adapter queue depth

When configuring the host bus adapter, the Execution Throttle or queue depth should be set to 255 as to not gate the driver module queue depth. Depending on firmware version, the Execution Throttle variable in QLogic cards is unavailable because it has been deprecated in favor of the variable being set in the driver module. The queue depth variable within the VMkernel driver module loaded for each HBA in the system and DSNRO ultimately regulate the host queue depth.

4.2 Storage driver queue depth and timeouts

The VMkernel driver module ultimately regulates the queue depth for the HBA if it needs to be changed. See appendix A for information about determining the appropriate queue depth.

In addition to setting the queue depth in the driver module, the disk timeouts must also be set within the same command. These timeouts need to be set for the ESXi host to properly survive an SC Series controller failover. To configure these settings, refer to the section, *Adjust Queue Depth for QLogic, Emulex, and Brocade HBAs*, in the document, *vSphere Troubleshooting* in the [VMware vSphere Documentation](#).

Caution: Before running the following commands, refer to the latest documentation from VMware for the latest information.

4.2.1 Fibre Channel HBAs

For each of these adapters, the method to set the driver queue depth and timeouts uses the following general steps:

1. Locate the appropriate driver name for the module that is loaded:

- For QLogic, enter:

```
esxcli system module list |grep ql
```

- For Emulex, enter:

```
esxcli system module list |grep lpfc
```

Depending on the HBA model, the output could be similar to the following:

- QLogic: qla2xxx or qlnativefc
- Emulex: lpfc820

Note: The following steps contain example module names. The actual module names should be acquired when completing step 1.

2. Set the driver queue depth and timeouts using the esxcli command:

- For QLogic, enter:

```
esxcli system module parameters set -m qlnativefc -p "ql2xmaxqdepth=255  
ql2xloginretrycount=60 qlport_down_retry=60"
```

- For Emulex, enter:

```
esxcli system module parameters set -m lpfc820 -p "lpfc_devloss_tmo=60  
lpfc_lun_queue_depth=254"
```

Note: In certain multipathing configurations, the qlport_down_retry value may be set lower to decrease failover times between paths if one of the paths fails.

3. Reboot the ESXi host for these changes to take effect.
4. To verify the settings, use the following command:

```
esxcli system module parameters list -m=module(i.e. -m=qla2xxx)
```

4.2.2 Software iSCSI initiator

Similarly, for the software iSCSI initiator, complete the following steps:

1. Set the queue depth to 255 (example shown):

```
esxcli system module parameters set -m iscsi_vmk -p iscsivmk_LunQDepth=255
```

2. If the login timeout is not already set to 5 seconds (the default), change it to 5 seconds (example shown):

- Determine the iSCSI adapter name:

```
esxcli iscsi adapter list
```

- Set the login timeout parameter:

```
esxcli iscsi adapter param set -A=vmhba37 -k>LoginTimeout -v=5
```

Note: The recommended LoginTimeout value has recently changed from 60 back to the VMware default of 5. If the ESXi host connects to both PS Series (60 recommended) and to SC Series (5 recommended), the timeouts should be individually set at the discovery address level within the ESXi iSCSI initiator.

3. Reboot the ESXi host for the change to take effect.
4. To verify the settings, use the following commands:

```
esxcli system module parameters list -m iscsi_vmk
esxcli iscsi adapter param get -A=vmhba37
```

Note: In earlier versions of ESXi 5.x, the option to set the login timeout parameter is not available. To enable the login timeout parameter, it may require applying the patch as described in [VMware KB article 2007680](#).

4.2.3 SAS HBAs

For SC Series arrays that have SAS front-end ports, use the following recommendations.

For the **mpt3sas**, use all the default driver module settings.

For the **lsi_msgpt3**, change the following module parameter setting as recommended by the Dell KB article [SC Storage Customer Notification: Driver Compatibility with Front End SAS Connectivity](#).

```
esxcli system module parameters set -p issue_scsi_cmd_to_bringup_drive=0 -m
lsi_msgpt3
```

Caution: The driver module in use differs between ESXi versions. Review the Dell KB article [Preparing VMware ESXi Hosts to Attach to SCv20x0, SCv30x0, SC4020, SC5020 SAS Arrays](#).

When using lsi_msgpt3 driver versions greater than 12, the `issue_scsi_cmd_to_bringup_drive` must be modified as shown above. Review the VMware KB article [Storage Center systems with Front End SAS connectivity show lun capacity OMB \(67032\)](#) for more information.

4.3 Adjusting settings for permanent device loss conditions

When using vSphere High Availability (HA), it is a best practice to modify the following host settings in dealing with permanent device loss (PDL) conditions.

- From within the host advanced system settings, apply the following modifications:
 - VMkernel.Boot.terminateVMonPDL = Yes/True (Default = No/False, Reboot required)

This setting automatically terminates a virtual machine that resides on a datastore in a PDL condition. For example, if a storage administrator accidentally removes a path mapping to a single host causing a PDL condition. This setting allows HA to terminate that virtual machine, and restart it on a different host within the cluster.

- vSphere 5.5: Disk.AutoremoveOnPDL = 0 (Not the default advanced setting)
- vSphere 6.0+: Disk.AutoremoveOnPDL = 1 (default advanced setting)

This setting prevents a disk device entering a PDL state from being automatically removed from a host, preventing an inadvertently removed device from being treated as a new device.

- From within the HA cluster advanced options, add the following configuration parameter:
 - das.maskCleanShutdownEnabled = True (Default setting = True)

This setting instructs the fault domain manager (FDM) to presume a virtual machine should be restarted when its home datastore is not accessible.

For more information about these settings, refer to the following:

- [vSphere Availability guide](#)
- [PDL Conditions and High Availability](#)
- [PDL AutoRemove feature in vSphere 5.5](#)

4.4 Modifying the VMFS queue depth for virtual machines (DSNRO)

Disk scheduled number requests outstanding (DSNRO) is an advanced setting within each ESXi host that controls the queue depth at the datastore level.

DSNRO is a value that can be increased or decreased depending on how many virtual machines are to be placed on each datastore or based on the VM I/O requirements. This queue depth limit is only enforced when more than one virtual machine per host is active on that datastore. For example, if the value is set to default, the first virtual machine active on a datastore will have its queue depth limited only by the VMkernel driver module. When a second, third, or fourth virtual machine is added to the datastore, during contention the limit will be enforced to the maximum queue depth of 32 (or as modified).

Note: Previous to ESXi 5.5, DSNRO was a global setting named Disk.SchedNumReqOutstanding that applied to all datastores. Modifying DSNRO for an individual datastore was not possible.

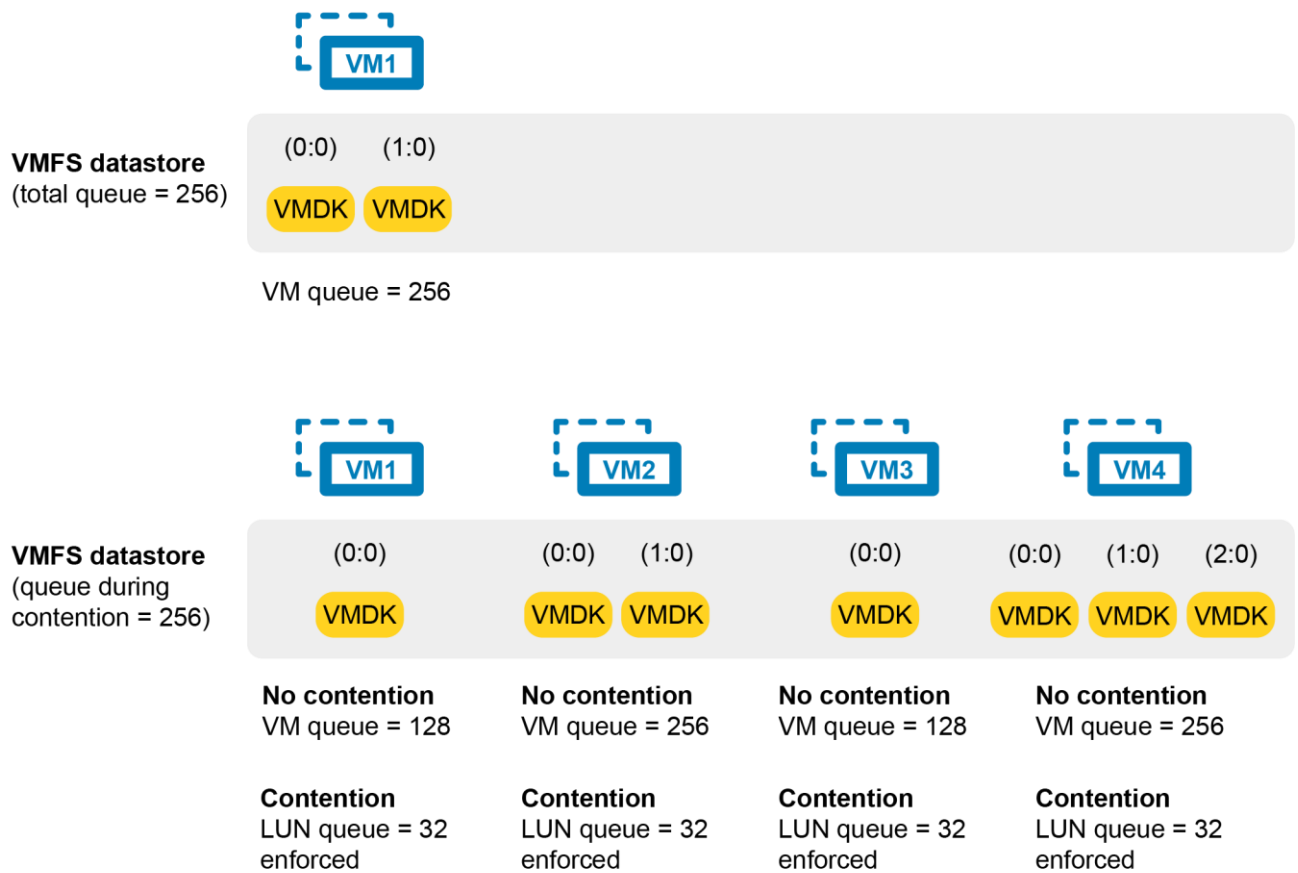


Figure 2 Example queue utilization with the DSNRO set to 32.

Note: The DSNRO limit does not apply to volumes mapped as raw device mappings (RDMs). Each RDM will have its own queue.

The DSNRO setting can be modified on a per-datastore basis using the command line:

```
esxcli storage core device set -d <naa.dev> -O <value of 1-256>
```

Note: This setting allows fine-tuning of DSNRO on a per-volume basis, however it must be set on each datastore (and each host) if the queue depth is greater than 32.

To globally set the DSNRO, the easiest method is to use a scripting tool such as the VMware PowerCLI utility. The following example script demonstrates setting the DSNRO to 64 globally across all datastores, on each host. Remember that any volume added after this script is ran would need to be changed manually or by rerunning the script.

Example PowerCLI Script (dsnro.ps1):

```
#Connect to vCenter. Change server, user, and password credentials below
Connect-VIServer -Server 'vCenter_Server_IP_or_FQDN' -User
'administrator@vsphere.local' -Password 'thepassword'

#Change this variable to the desired dsnro (Default=32 Max=64)
$dsnro = 64

#Retrieve a list of ALL ESXi hosts from the vCenter server
$esxhosts = get-vmhost

#Cycle through each host retrieving all storage devices associated with that
host
foreach ($hostname in $esxhosts)
{
    $esxcli = Get-EsxCli -VMHost $hostname
    $devices = $esxcli.storage.core.device.list()
    foreach ($device in $devices)
    {
        if ($device.Vendor -like "COMPELNT")
        {
            $esxcli.storage.core.device.set($false, $null, $device.Device, $null,
            $null, $null, $null, $null, $null, $null, $null, $dsnro,$null,$null)
            $esxcli.storage.core.device.list()
        }
    }
}
```

As a best practice for modifying any host settings, run tests to determine the impact of changing this variable beforehand. Once successful tests have been completed, as a precautionary measure, it is recommended only to run this script during a maintenance window. Since DSNRO helps to ensure fairness across the virtual machines residing on a datastore, modifying this value could lead to individual virtual machines monopolizing disk resources.

More information about the DSNRO can be found in the following documentation:

- “Change Maximum Outstanding Disk Requests in the vSphere Web Client” in the appropriate *vSphere Troubleshooting guide*: [VMware vSphere documentation](#)
- [Setting the Maximum Outstanding Disk Requests for virtual machines \(1268\)](#) in the VMware Knowledge Base

4.5 Adaptive queue depth

At times of high congestion, VMware has an adaptive queue depth algorithm that can be enabled with the `QFullSampleSize` and `QFullThreshold` variables. These variables aid in relieving the congestion by dynamically reducing and increasing the logical unit number (LUN) queue depth. Due to the architecture of SC Series storage, enabling these settings is not recommended unless under the guidance of Dell Support. For more information, see [Controlling LUN queue depth throttling in VMware ESX/ESXi](#) in the VMware Knowledge Base.

4.6 Modifying the guest operating system queue depth

The queue depth can also be set within the guest operating system if needed. Windows operating systems have a default queue depth of 32 set for each vSCSI controller, but can be increased up to 128 if necessary. The method to adjust the queue depth varies between operating systems. Examples are shown in section 4.6.1. Extra vSCSI adapters can be added to a virtual machine to increase the total queue depth available to the virtual machine.

4.6.1 Windows Server

To adjust the queue depth for the LSI® Logic drivers, add or modify the following registry keys:

1. Before beginning, back up the registry.
2. Use `regedit` to add the following keys:

- a. For LSI Logic Parallel (LSI_SCSI):

Windows Registry Editor Version 5.00

```
[HKLM\SYSTEM\CurrentControlSet\Services\LSI_SCSI\Parameters\Device]
"DriverParameter"="MaximumTargetQueueDepth=128;"
; The semicolon is required at the end of the queue depth value
"MaximumTargetQueueDepth"=dword:00000080
; 80 hex is equal to 128 in decimal
```

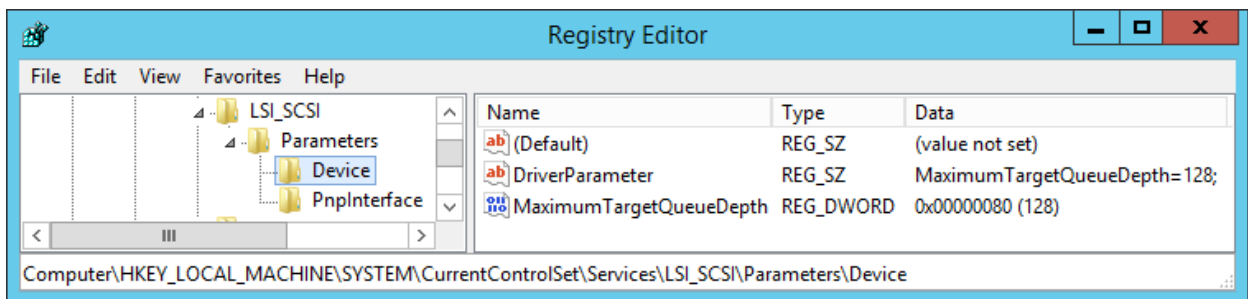


Figure 3 Registry settings for the LSI Logic Parallel vSCSI adapter

- b. For LSI Logic SAS (LSI_SAS):

Windows Registry Editor Version 5.00

```
[HKLM\SYSTEM\CurrentControlSet\Services\LSI_SAS\Parameters\Device]
"DriverParameter"="MaximumTargetQueueDepth=128;"
; The semicolon is required at the end of the queue depth value
"MaximumTargetQueueDepth"=dword:00000080
; 80 hex is equal to 128 in decimal
```

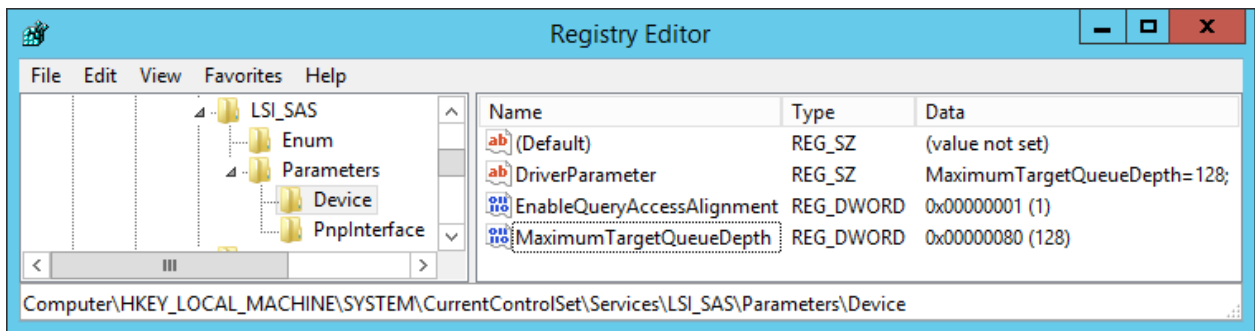


Figure 4 Registry setting for the LSI Logic SAS vSCSI Adapter

- c. For VMware Paravirtual SCSI (PVSCSI), the Paravirtual adapter is different from the LSI vSCSI adapters in that its queue depth can be adjusted higher: Up to 256 for devices and 1024 for the adapter.

For more information, see the VMware KB article, [Large-scale workloads with intensive I/O patterns might require queue depths significantly greater than Paravirtual SCSI default values.](#)

3. Reboot the virtual machine.

Note: See the [VMware Knowledge Base](#) for the most current information about setting the queue depth with different vSCSI controllers or operating systems.

4.7 Setting operating system disk timeouts

For each operating system, if VMware tools are not installed, the disk timeouts must be set to 60 seconds for the operating system to handle storage controller failovers properly.

Examples of setting the operating-system timeouts can be found in the section, “Set Timeout on Windows Guest OS” in the *vSphere Storage Guide* at [VMware vSphere documentation](#).

The general steps to set the disk timeout within Windows and Linux are listed in the following sections.

4.7.1 Windows

1. Back up the registry.
2. Using the Registry Editor, modify the following key.

Windows Registry Editor Version 5.00

```
[HKLM\SYSTEM\CurrentControlSet\Services\Disk]
"TimeoutValue"=dword:0000003c
; 3c in hex is equal to 60 seconds in decimal
```

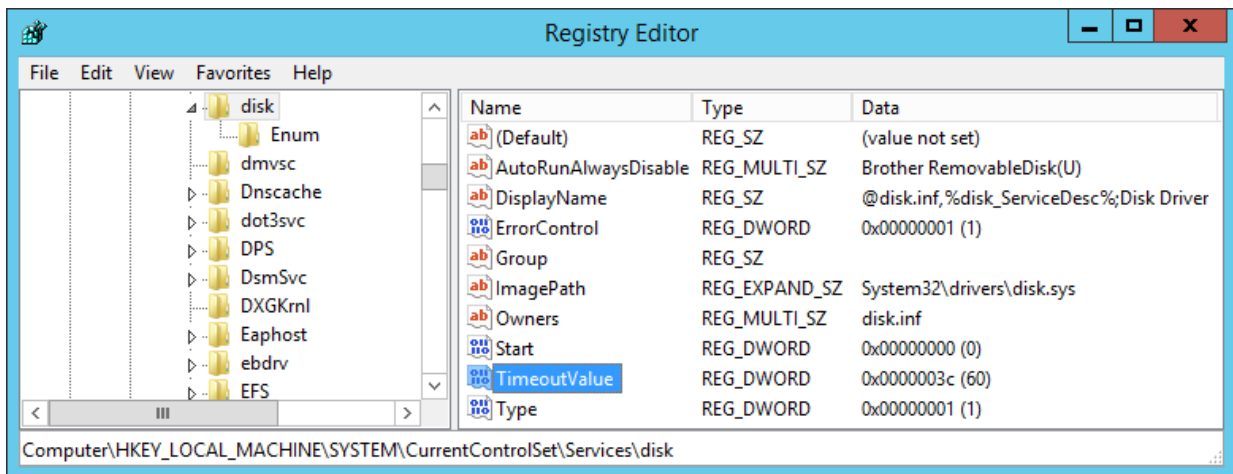


Figure 5 Registry key to set the Windows disk timeout

3. Reboot the virtual machine.

Note: This registry value is automatically set when installing VMware Tools. For more information, see [Inconsistent Windows virtual machine performance when disks are located on SAN datastores](#) in the VMware Knowledge Base.

4.7.2 Linux

For more information about setting disk timeouts in Linux, refer to the VMware Knowledge Base article, [Increasing the disk timeout values for a Linux 2.6 virtual machine](#).

Caution: For Red Hat Enterprise Linux 7.x virtual machines, verify the disk timeouts are correctly set to 60 seconds. For more information, see the Red Hat knowledge base (login required), article 1578643: [The udev rules for SCSI timeout are missing in open-vm-tools package in Red Hat Enterprise Linux 7](#).

5 Guest virtual SCSI adapter selection

When creating a new virtual machine, there are four types of virtual SCSI (vSCSI) controllers to choose from. Based on the operating system selected, vSphere will automatically recommend and select a SCSI controller that is best suited for that particular operating system. The best practice is to follow the client recommendation. The nuances of each adapter are described in the following subsections.

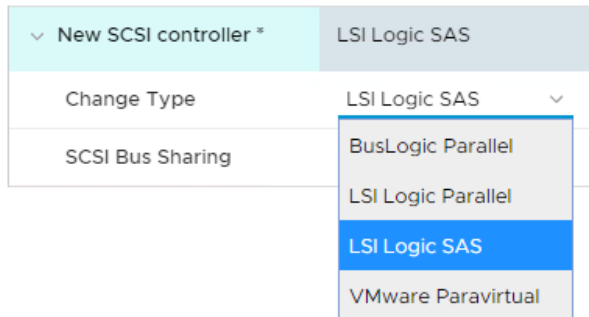


Figure 6 vSCSI adapter selection

5.1 BusLogic Parallel

This vSCSI controller is used for older operating systems. Due to the queue depth limitations of this controller, it is not recommended unless it is the only option available for that particular operating system. Certain versions of Windows issue only enough I/O to fill a queue depth of one.

5.2 LSI Logic Parallel

Because many operating systems support this vSCSI adapter, it is recommended for virtual machines that do not support the LSI Logic SAS adapter.

5.3 LSI Logic SAS

This vSCSI controller is available for virtual machines with hardware versions 7 and later. It also has similar performance characteristics of the LSI Logic Parallel. This adapter adds support for SCSI-3 reservations, which are required for Microsoft Cluster Services (MSCS). Some operating system vendors are gradually withdrawing support for SCSI in favor of SAS, making the LSI Logic SAS controller a good choice for future compatibility.

5.4 VMware Paravirtual

This vSCSI controller is a high-performance adapter that can result in greater throughput and lower CPU utilization. More information about the usage and limitations of this adapter can be found in the section, *About VMware Paravirtual SCSI Controllers*, in the *vSphere Virtual Machine Administration Guide* in the [VMware vSphere documentation](#).

6 Mapping volumes to an ESXi server

Within the SC Series, mapping is the process of presenting a volume to a host. The following subsections describe basic concepts on how vSphere treats different scenarios.

6.1 Basic volume mapping concepts

When sharing volumes between ESXi hosts for vMotion, HA, and DRS, for consistency it is recommended that each volume is mapped to clustered ESXi hosts using the same LUN.

For example:

- There are three ESXi hosts named ESXi1, ESXi2, and ESXi3
- A new volume is created named "LUN10-vm-storage"
- This volume must be mapped to each of the ESXi hosts as the same LUN:

Volume: "LUN10-vm-storage" → Mapped to ESXi1 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi2 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi3 -as- LUN 10

6.2 Basic SC Series volume mappings

In SCOS versions 5.x and later, the mapping process is automated by creating a server cluster object. This feature allows the volume to be mapped to multiple ESXi hosts simultaneously, automatically keeping the LUN numbering consistent for all the paths.

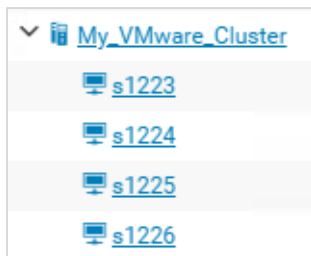
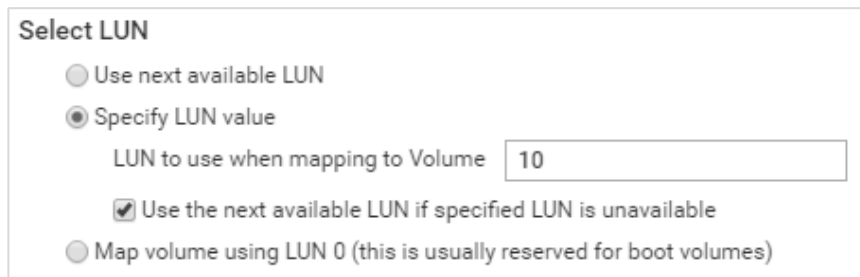


Figure 7 Example of a server cluster object

As an added benefit, when a new ESXi host is placed into the server cluster, all the existing volume mappings assigned to the cluster object are applied. Meaning that if the cluster has 100 volumes mapped to it, presenting them to a new ESXi host is as simple as adding it to the cluster object.

Similarly, if the host is removed from the server cluster, the cluster mappings are removed. All I/O must be stopped from the host before volumes are removed. Only volumes that are mapped to an individual host, such as the boot volume, will remain once a host is removed from the server cluster.

In the mapping wizard, the system can auto select the LUN number, or a preferred LUN number can be manually specified from the advanced settings screen shown in Figure 8.



The screenshot shows a dialog box titled "Select LUN". It contains four radio button options: "Use next available LUN", "Specify LUN value" (which is selected), "Use the next available LUN if specified LUN is unavailable" (which has a checked checkbox), and "Map volume using LUN 0 (this is usually reserved for boot volumes)". Below the "Specify LUN value" option, there is a text input field with the value "10" entered.

Figure 8 Manually specifying a LUN in the advanced settings screen

This advanced option allows administrators who already have a LUN numbering scheme to continue using it. However, if a LUN is not manually specified, the system automatically selects a LUN for each volume incrementally starting at LUN 1.

Timesaver: When naming volumes from within the SC Series user interface, it may be helpful to specify the LUN number as part of the volume name. This naming helps to quickly identify which volumes are mapped using each LUN.

6.3 Multipathed volume concepts

If there are ESXi hosts that have multiple initiator ports, ESXi has integrated functionality to provide native multipathing of volumes over multiple supported protocols.

Building on the previous example, here is an example of multipathing mappings:

Volume: "LUN10-vm-storage" → Mapped to ESXi1/HBA1 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi1/HBA2 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi2/HBA1 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi2/HBA2 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi3/HBA1 -as- LUN 10

Volume: "LUN10-vm-storage" → Mapped to ESXi3/HBA2 -as- LUN 10

Note: With older versions of ESXi, if the LUN number is not consistent between multiple hosts or multiple HBAs, VMFS datastores may not be visible to all nodes.

Keep in mind that when a volume uses multiple paths, both the ESXi initiators are mapped from the same controller, through different front-end ports. For example:

"LUN10-vm-storage" → Controller1/Port1 → FC-Switch-1 → Mapped to ESXi1/HBA1 as LUN 10

"LUN10-vm-storage" → Controller1/Port2 → FC-Switch-2 → Mapped to ESXi1/HBA2 as LUN 10

If a different volume is active on the second controller, it may be mapped such as:

"LUN20-vm-storage" → Controller2/Port1 → FC-Switch-1 → Mapped to ESXi1/HBA1 as LUN 20

"LUN20-vm-storage" → Controller2/Port2 → FC-Switch-2 → Mapped to ESXi1/HBA2 as LUN 20

When configuring multipathing, remember that a volume cannot be mapped to both controllers simultaneously, because a volume can only be active on one controller at a time.

6.4 Multipathed SC Series volumes

When multipathing SC Series volumes, the process is automated. Selecting the correct operating system in the server properties screen can prevent many of the common mapping errors. Based on the operating system selected, SC Series correctly maps volumes by applying a set of rules to the server that are unique to each operating system.

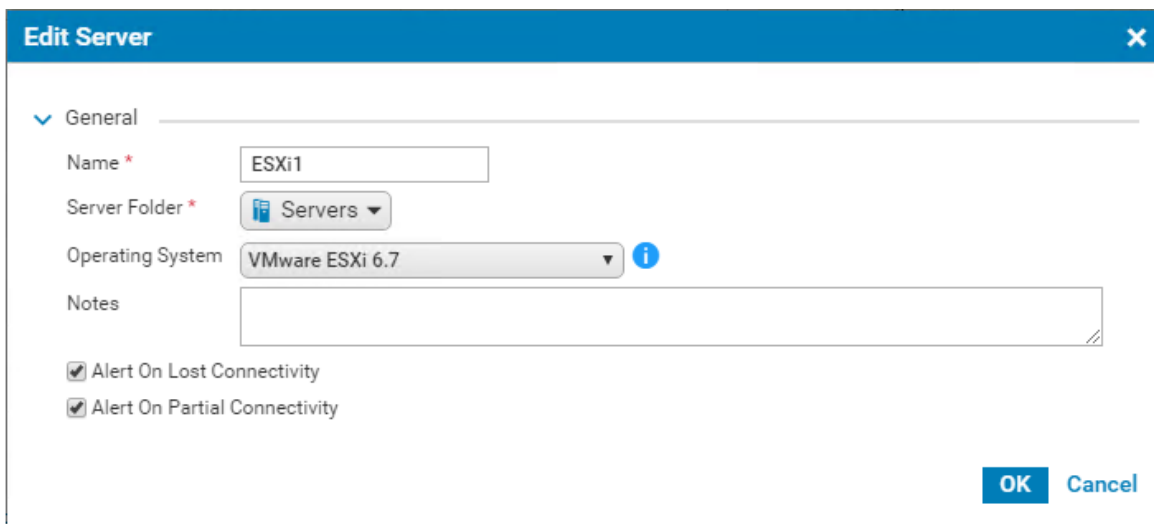


Figure 9 Server operating system selection

Multipathing to an ESXi host is automatic when the server object has more than one HBA or iSCSI initiator ports assigned to it. In other words, the advanced options must be used if the server does not need a volume multipathed.

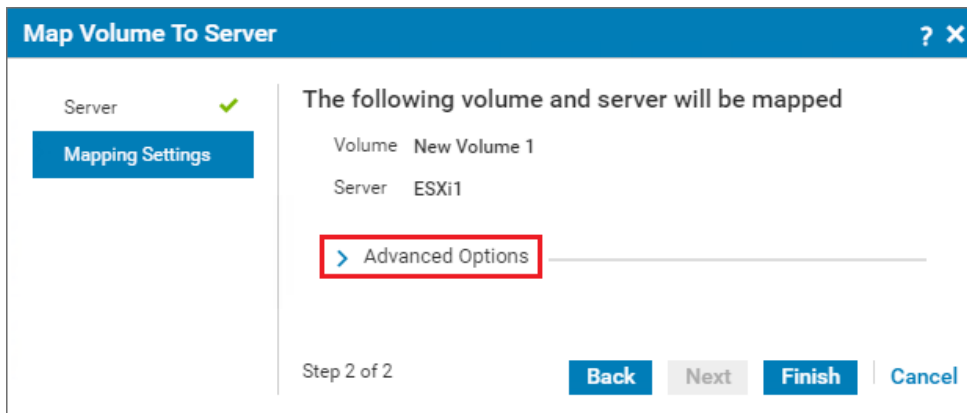


Figure 10 Advanced server mapping options

Table 3 Advanced mapping options for an ESXi host

Function	Description
Select LUN	Clear the Use next available LUN option to manually specify the LUN. If this box is not checked, the system automatically assigns the next available LUN.
Restrict Mapping Paths	Use this option when a volume only needs to be mapped to a specific HBA in the ESXi host.
Configure Multipathing	This option designates how many of the SC Series front-end ports allowed for the volume to be mapped through. For example, if each controller has four front-end ports, selecting unlimited maps the volume through all four. Selecting two only uses two of the four front-end ports. The system automatically selects the two front-end ports with the fewest mappings.
Configure Volume Use	VMFS does not recognize read-only mappings, so this option should not be used.

6.5 Configuring the VMware iSCSI software initiator for a single path

Although it is not recommended, for instances where iSCSI multipathing cannot be configured, the steps required for a single-path iSCSI configuration are as follows.

From within the VMware vSphere Client:

1. In the ESXi host **Security Profile** > **ESXi firewall**, enable the **Software iSCSI Client**.
2. Add a VMkernel port to a virtual switch assigned to the physical NIC for iSCSI (see Figure 11).

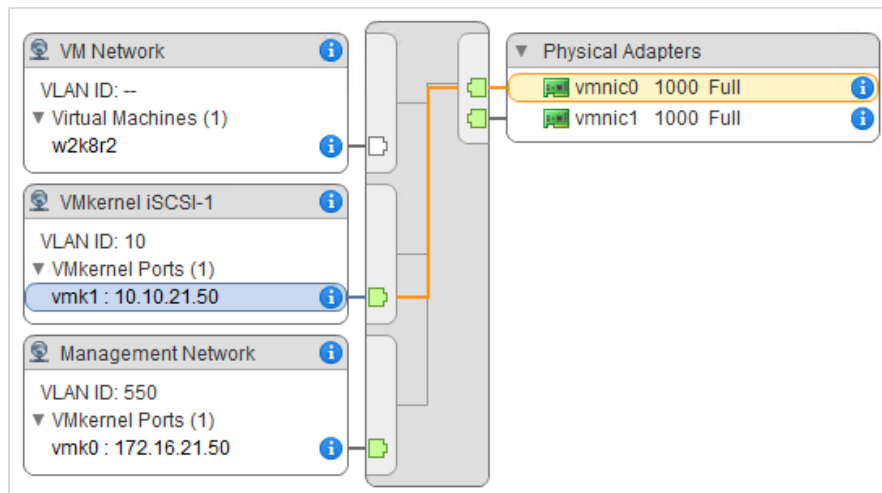


Figure 11 Configuring the VMkernel port

3. From the **Storage Adapters** configuration screen, click the plus sign to add an adapter, select **Software iSCSI Adapter**, and click OK.
4. Within the **Storage Adapters** area, highlight the iSCSI software adapter (such as vmhba38).
5. To add the SC Series front-end ports, select **Targets**, then select the **Dynamic Discovery** tab. Click **Add** and enter the iSCSI control port IP address, labeled as the well-known IP address for the fault domain. If virtual ports are not enabled for iSCSI front-end ports, each of the iSCSI adapter IP addresses for the system must be added manually.
6. Rescan the iSCSI initiator.
7. From DSM:
 - a. Create a server object for the ESXi host.
 - b. Map a volume to the ESXi host.
8. From within the VMware vSphere client, browse the **Storage Adapters** section and rescan the iSCSI HBA for new volumes.

6.6 Configuring the VMware iSCSI software initiator for multipathing

To configure the VMware iSCSI software initiator for multipathing, see sections “Configuring Software iSCSI Adapter” and “Multiple Network Adapters in iSCSI Configuration” in the *vSphere Storage Guide* at [VMware vSphere documentation](#).

For users with previous experience configuring iSCSI for multipathing, here are a few key points for a successful configuration:

- Verify that there is one VMkernel interface for each physical NIC to be used for storage traffic, following the virtual switch port binding recommendations in section 6.6.1.
- Adjust the failover order on each VMkernel interface for a 1:1 VMkernel to physical NIC ratio.
- Add both VMkernel interfaces to the iSCSI software adapter network port binding (see Figure 12). If the prerequisite failover orders have not been set, the vSphere client will not allow the operation.
- Rescan the iSCSI software adapter for new volumes.
- From within the Dell Storage Manager Client, create the server object for the host.

The screenshot shows the configuration for the iSCSI Software Adapter 'vmhba38'. The adapter is online and has the iqn.1998-01.com.vmware:tssrv248-2b03cc0b. The 'Network Port Binding' tab is selected, showing two port groups: 'VMkernel iSCSI-1 (vSwitch0)' and 'VMkernel iSCSI-2 (vSwitch0)'. Both are bound to VMkernel adapters 'vmk1' and 'vmk2' respectively, with a 'Compliant' status. The path status for both is 'Not used', and they are connected to physical network adapters 'vmnic0' and 'vmnic1'.

Port Group	VMkernel Ad...	Port Group Policy	Path Status	Physical Network Adapter
VMkernel iSCSI-1 (vSwitch0)	vmk1	Compliant	Not used	vmnic0 (1 Gbit/s, Full)
VMkernel iSCSI-2 (vSwitch0)	vmk2	Compliant	Not used	vmnic1 (1 Gbit/s, Full)

Figure 12 Binding multiple VMkernel ports to the software iSCSI initiator

6.6.1 VMkernel port binding considerations with software iSCSI

When using the software iSCSI initiator provided within an ESXi host, careful consideration needs to be given to the appropriate virtual switch configurations adhering to VMware iSCSI port-binding requirements. The number of fault domains, subnets, and IP configuration need to be carefully examined to avoid configuration errors. If iSCSI front-end ports are configured in a single subnet and single fault domain, the vmnics must reside within a single vSwitch to use VMkernel port binding.

Note: The most common environment in which a single fault domain is used for the iSCSI network is a redundant configuration with a single VLAN.

For example, with subnet 192.168.0.x and subnet mask 255.255.255.0, the vSwitch would look like Figure 13.

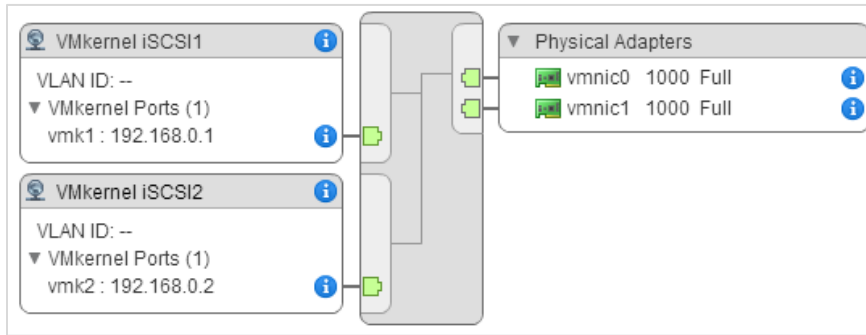


Figure 13 iSCSI ports and NICs on a single vSwitch/single subnet (VMkernel port binding allowed)

Note: Using port binding is not recommended when the VMkernel ports are on different networks as shown in Figure 14, because it may cause long rescan times and other problems. See [Considerations for using software iSCSI port binding in ESX/ESXi](#) in the VMware Knowledge Base.

In configurations using multiple fault domains, with the iSCSI front-end ports configured in a multiple subnet and multiple fault domain configurations, ensure either:

- VMkernel ports and vmnics reside in separate vSwitches (see Figure 14)
- OR
- Failover order ensures one VMkernel per physical NIC in a single vSwitch

Due to the network separation, VMkernel port binding must **not** be used.

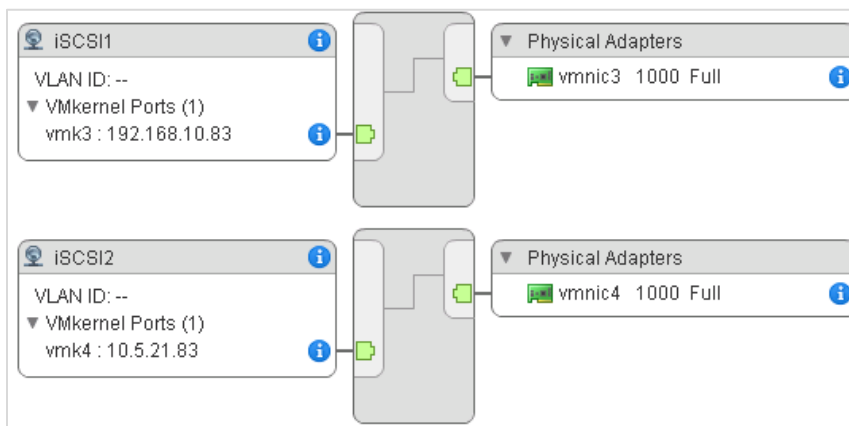


Figure 14 iSCSI VMkernel ports and NICs on multiple vSwitches/multiple subnets (port binding must not be used).

6.7 iSCSI port multi-VLAN configuration recommendations

For SC Series systems with iSCSI capabilities, SCOS 6.5 and later has multiple-VLAN support for diverse network configurations and multitenant environments. This multi-VLAN support allows iSCSI storage I/O to be separated between vSwitches for customers using software iSCSI initiators within the guest. It also allows admins to isolate specific ESXi host-cluster VMkernel ports to their own dedicated iSCSI storage VLAN.

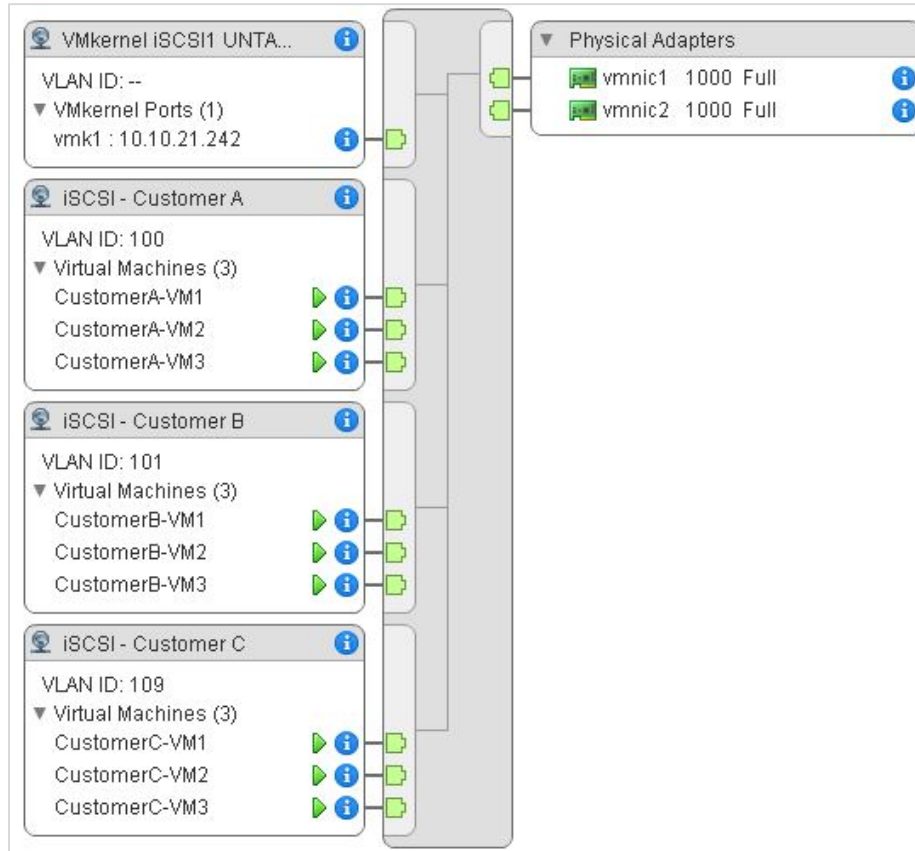


Figure 15 VLANs assigned to vSwitches isolating in-guest iSCSI initiator traffic for customers

Since iSCSI traffic is not encrypted in its plain form, it is a best practice to isolate that traffic for security purposes. A common misconception is that CHAP encrypts iSCSI traffic, but it only provides authentication for the connection to prevent unauthorized access.

Note: In older versions of DSM, the configuration of iSCSI VLANs is functionality that can only be configured through the traditional Dell Storage Manager Client.

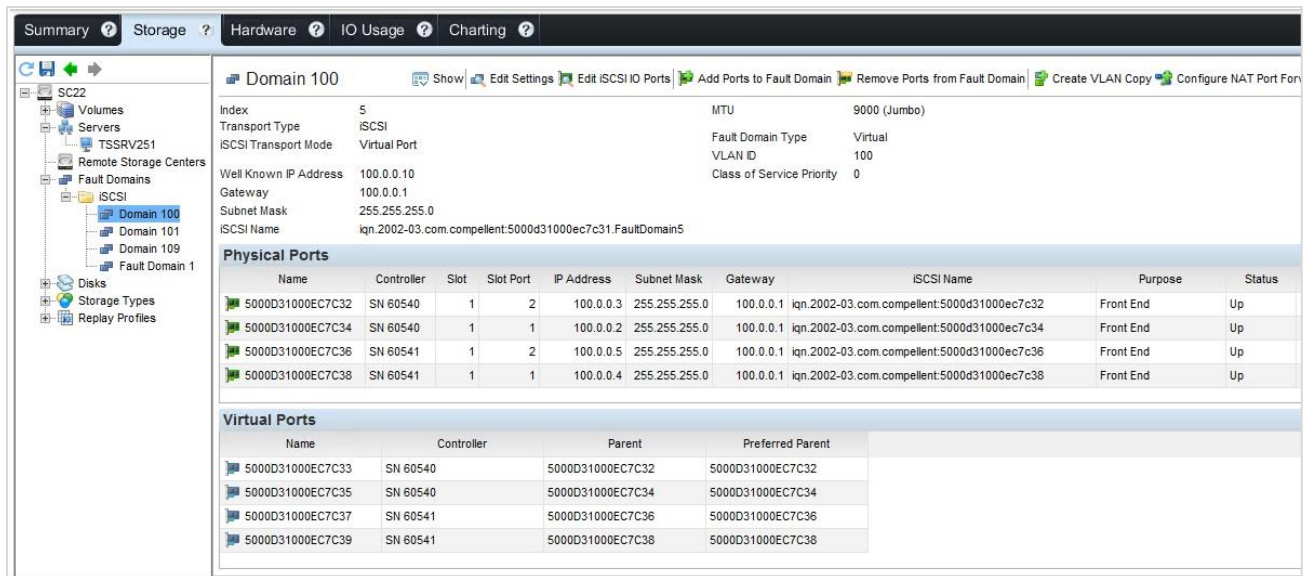


Figure 16 Configuration of VLANs within the Dell Storage Manager (Enterprise Manager) client

Note: For added flexibility, Jumbo frames can be enabled on a per-fault-domain basis.

6.8 Configuring the FCoE software initiator for multipathing

When using the ESXi software FCoE, volumes are assigned to the VMW_SATP_LOCAL by default. In some instances, this default policy will cause paths to go unclaimed. To remedy this situation, it may be necessary to adjust the claim rules on each ESXi host so that FCoE volumes are correctly assigned to the VMW_SATP_ALUA.

```
esxcli storage nmp satp rule add -R fcoe -s VMW_SATP_ALUA
```

6.9 VMware multipathing policies

When configuring the path-selection policy of each datastore or volume, administrators have the choice of round robin, fixed, or most recently used. The default path selection policy for the SC Series system depends on the SCOS version and the associated SATP module detected. Round robin is highly recommended for ease of management if there are no technical reasons against using it (such as with Microsoft failover clusters).

6.9.1 Round robin policy for standard volumes

The round robin path selection policy uses automatic path selection and load balancing to rotate I/O through all paths. Round robin load balancing does not aggregate the storage link bandwidth. It merely distributes the load for the datastore volume in bursts evenly and sequentially across paths in an alternating fashion.

Using round robin reduces the management headaches of manually balancing the storage load across all storage paths as with a fixed policy. However, there are certain situations where using round robin does not make sense. For instance, it is not considered a best practice to enable round robin between an iSCSI path and Fibre Channel path. It is also not best practice to balance the load between an 8 Gb FC and a 16 Gb FC path. With round robin, care must be taken to ensure all the paths included are identical in type, speed, and have the same queue depth settings.

Here is an example of what happens during a path failure when using round robin:

1. Load is distributed evenly between HBA1 and HBA2
2. HBA1 loses connectivity; HBA2 will assume all I/O load
3. HBA1 resumes connectivity; load is distributed evenly again between both

Multipathing Details

Device: COMPELNT Fibre Channel Disk (naa.6000d3100002b900000000000000deb9f)

Multipathing Policies Edit Multipathing...

▼ Path Selection Policy	
Path Selection Policy	Round Robin (VMware)
Preferred Path	
Storage Array Type Policy	VMW_SATP_ALUA

Paths

Owner Plugin: NMP

▼ Paths

Refresh Enable Disable

Runtime Name	Status	Target	LUN	Preferred
vmhba3:C0:T3:L100	◆ Active (I/O)	50:00:d3:10:00:02:b9:02 50:00:d3:...	100	
vmhba2:C0:T3:L100	◆ Active (I/O)	50:00:d3:10:00:02:b9:02 50:00:d3:...	100	
vmhba3:C0:T2:L100	◆ Active (I/O)	50:00:d3:10:00:02:b9:02 50:00:d3:...	100	
vmhba2:C0:T2:L100	◆ Active (I/O)	50:00:d3:10:00:02:b9:02 50:00:d3:...	100	

Figure 17 Example of a datastore path selection policy set to round robin

6.9.1.1 Setting round robin to be the default path selection policy

The round robin path selection policy (PSP) should be set to the default using the following command. After creating the claim rule and rebooting, all SC Series volumes and protocol endpoints will acquire this policy.

```
esxcli storage nmp satp rule add -s VMW_SATP_ALUA -V COMPELNT -P VMW_PSP_RR -o
disable_action_OnRetryErrors -e "Dell EMC SC Series Claim Rule" -O
"policy=iops;iops=3"
```

Caution: With ESXi 6.7 and later, VMware has set the default SATP option for **action_OnRetryErrors** from **off** to **on**, which may cause premature APD events during storage controller failover scenarios. See [VMware KB article 67006](#) for more information.

Caution: Host profiles may overwrite these claim rule options. In addition, the round robin path selection policy may be unsupported for use with Microsoft Clustering Services depending on the version of ESXi. Check the current support status.

6.9.1.2 Conditions for path changes

When using the round robin path selection policy, the native multipathing plug-in (NMP) has custom properties that can be tuned to optimize traffic flow between each of the paths. By default, the round robin PSP will switch paths every 1,000 I/Os or 10 MB, whichever comes first. Dell internal lab testing has shown that modifying the IOPS value can have a noticeable performance impact based on the I/O patterns. Because the optimum tuning value for each datastore depends on the data patterns issued by the guests, thorough testing is recommended to prevent negative performance. When beginning testing, a good starting point is to set the IOPS to 3 as shown in the following.

To set the values for a datastore, use the following steps:

1. Find the datastore device NAA identifier.

```
~ # esxcli storage nmp device list
naa.6000d31000ed1f0000000000000000179
  Device Display Name: COMPELNT Fibre Channel Disk
(naa.6000d31000ed1f0000000000000000179)
  Storage Array Type: VMW_SATP_ALUA
  Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=on;
{TPG_id=61475,TPG_state=AO}{TPG_id=61479,TPG_state=AO}{TPG_id=61476,TPG_st
ate=AO}{TPG_id=61480,TPG_state=AO}}
  Path Selection Policy: VMW_PSP_FIXED
  Path Selection Policy Device Config:
{preferred=vmhba3:C0:T6:L120;current=vmhba3:C0:T6:L120}
  Path Selection Policy Device Custom Config:
Working Paths: vmhba3:C0:T6:L120
Is USB: false
```

2. If not set previously, modify the disk device to use the round robin PSP.

```
~ # esxcli storage nmp device set --device=naa.xxx --psp=VMW_PSP_RR
```

3. Set the IOPS of the device to 3.

```
~ # esxcli storage nmp psp roundrobin deviceconfig set --device naa.xxx --  
type=iops --iops=3
```

Note: Monitor the vSphere host processor because changing these variables can affect its performance.

To automatically set the IOPS path change condition for all volumes mapped to an ESXi host, the claim rule can be modified by adding **-O "policy=iops;iops=3"**. For example:

```
esxcli storage nmp satp rule add -s VMW_SATP_ALUA -V COMPELNT -P VMW_PSP_RR -o  
disable_action_OnRetryErrors -e "Dell EMC SC Series Claim Rule" -O  
"policy=iops;iops=3"
```

Caution: This command contains two options that are case-sensitive. The lowercase **-o** is for the SATP option string, and an uppercase **-O** is for the PSP option string.

To remove a previously created claim rule, issue the following command:

```
esxcli storage nmp satp rule remove -s VMW_SATP_ALUA -V COMPELNT -P VMW_PSP_RR  
-o disable_action_OnRetryErrors -e "Dell EMC SC Series Claim Rule" -O  
"policy=iops;iops=3"
```

A reboot is required for this setting to take effect.

For more information and command examples for adjusting these settings, refer to the article, [Adjusting Round Robin IOPS limit from default 1000 to 1](#) in the VMware Knowledge Base.

6.9.2 Round robin policy for Live Volumes

The Live Volume Asymmetric Logical Unit Access (ALUA) feature was added in SCOS 7.3 and DSM 2018 following the T10 SCSI-3 specification SPC-3 for multiple operating systems. The feature allows the advertisement of MPIO paths to the primary Live Volume as optimal while MPIO paths to the secondary Live Volume are non-optimal. When used with the VMware round robin path selection policy (PSP), optimized paths will be used for read and write I/O when available. Non-optimized paths will be reserved for use in the event no optimal paths are available. If a Live Volume role swap or automatic failover occurs, the SC Series array will report the ALUA path state changes to the storage host upon request. The feature allows a round robin PSP to be used with Live Volume in either uniform or non-uniform storage presentations. The round robin PSP is easy to deploy, requires minimal administrative effort and documentation, and yields a good balance of storage port, fabric, and controller utilization.

For more implementation and best-practices information about the Live Volume ALUA feature, see the document, [Dell EMC SC Series: Synchronous Replication and Live Volume](#).

6.9.3 Fixed policy

While round robin is the recommended policy, if the fixed policy is used, it gives fine-tuned control over the flow of storage traffic. However, administrators must use caution to evenly distribute the load across all host HBAs, front-end ports, fabrics, and SC Series controllers.

When using the fixed policy, if a path fails, all the datastores using it as their preferred path will fail over to the secondary path. When service resumes, the datastores will resume I/O on their preferred path.

Here is an example of using the fixed policy:

1. HBA1 loses connectivity; HBA2 takes over its connections.
2. HBA1 resumes connectivity; HBA2 will fail its connections back to HBA1.

Multipathing Details

Device: COMPELNT Fibre Channel Disk (naa.6000d3100002b900000000000000deb9f)

Multipathing Policies Edit Multipathing...

Path Selection Policy	Fixed (VMware)
Preferred Path	vmhba3:C0:T3:L100
Storage Array Type Policy	VMW_SATP_ALUA

Paths

Owner Plugin: NMP

Refresh Enable Disable

Runtime Name	Status	Target	LUN	Preferred
vmhba3:C0:T3:L100	Active (I/O)	50:00:d3:10:00:02:b9:02 50:00:d3...	100	*
vmhba2:C0:T3:L100	Active	50:00:d3:10:00:02:b9:02 50:00:d3...	100	
vmhba3:C0:T2:L100	Active	50:00:d3:10:00:02:b9:02 50:00:d3...	100	
vmhba2:C0:T2:L100	Active	50:00:d3:10:00:02:b9:02 50:00:d3...	100	

Figure 18 Example of fixed datastore path selection policy set with a preferred path

6.9.4 Most recently used policy

The most recently used (MRU) PSP is used with active/passive arrays (to prevent path thrashing). MRU is not recommended or tested for use with SC Series storage because a volume can only be active on one controller at a time.

6.10 Multipathing using a fixed path selection policy

Keep in mind that with a fixed policy, only the preferred path transfers data. To distribute the I/O loads for multiple datastores over multiple HBAs, the preferred path must be set for each datastore consistently between each host. Here are some bad and good examples:

Example 1 (bad example):

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 10 (Active/Preferred)

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 10 (Standby)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 20 (Active/Preferred)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 20 (Standby)

This example would cause all I/O for both volumes to be transferred over HBA1.

Example 2 (good example):

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 10 (Active/Preferred)

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 10 (Standby)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 20 (Standby)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 20 (Active/Preferred)

This example sets the preferred path to distribute the load between both HBAs.

Although the fixed multipathing policy gives more fine-tuned control over pathing, manual validation is required to ensure that all paths have proportional amounts of traffic to each ESXi host.

6.11 Multipathing using a round robin path selection policy

When using round robin, it will provide path failure protection and will remove some of the guesswork of distributing load between paths manually. To reiterate from previous sections, when using round robin, be sure that the paths are of the same type, speed, and have the same queue depth setting.

Example 1:

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 10 (Active)

Volume: "LUN10-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 10 (Active)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA1 -as- LUN 20 (Active)

Volume: "LUN20-vm-storage" → Mapped to ESX1/HBA2 -as- LUN 20 (Active)

Storage traffic is evenly distributed between all HBAs.

6.12 Asymmetric logical unit access (ALUA) for front-end SAS

The ALUA protocol was designed for arrays that VMware classifies as asymmetrical storage systems. Historically, since the SC Series array is considered an active/active storage system where all the paths to standard volumes are always active (unless a path failed), ALUA was not necessary.

However, with the introduction of the SAS protocol to Dell Storage SCv2000 series arrays, ALUA became necessary for implementation into the front-end target ports for controller failover purposes. For example, for the path to the controller where the volume is active, ALUA will set the path state as active-optimized. For the path to the secondary controller, ALUA will set the path state to standby. During a controller failover, the ALUA path state will change such that the secondary controller will become the active-optimized path.

The other protocols supported by SC Series storage such as Fibre Channel, iSCSI, and FCoE, the pathing architecture remains mostly unchanged. Consistent with the former architecture, all the volume paths to the controller that the volume is active on will be labeled active-optimized. Because SC Series storage does not

proxy standard volume traffic between controllers, the paths to the second controller will be set to the standby state, and only change during a controller failover.

Note: Because of the ALUA protocol addition, the vSphere storage array type plug-in (SATP) module settings, such as configuring round robin as the default PSP, will need to be changed on each host when upgrading to SCOS 6.6 and later.

For more detailed information about ALUA in a vSphere environment, see the VMware vSphere Blog post, [Configuration Settings for ALUA Devices](#).

6.13 Unmapping volumes from an ESXi host

Within ESXi, VMware has added the ability to gracefully remove volumes before unmounting them to prevent an all-paths down (APD) state.

Before attempting this procedure, consult the section, “Performing Planned Storage Device Removal” in the *vSphere Storage Guide* at [VMware vSphere documentation](#).

1. Make note of the volume naa identifier. The naa ID will be referenced later.
2. From the datastore view, right-click the datastore and select **Unmount Datastore**

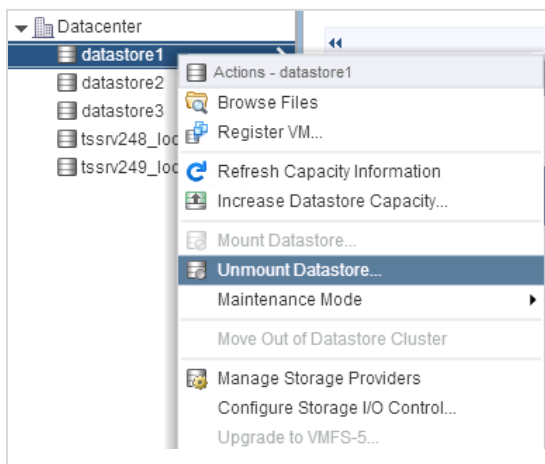
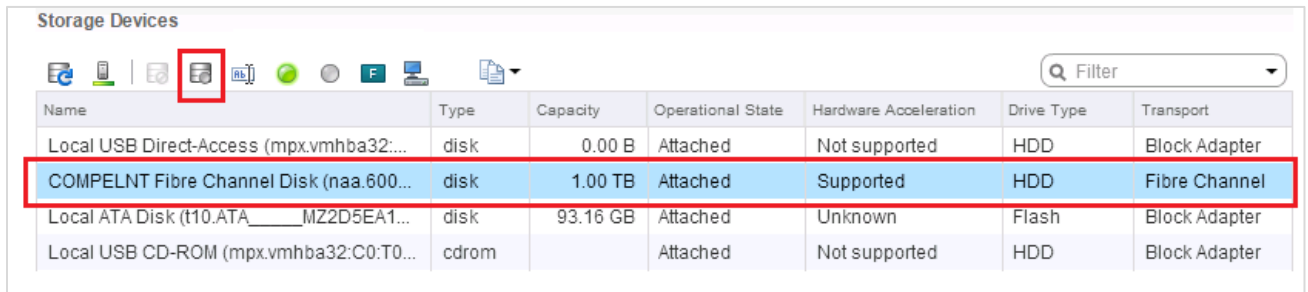


Figure 19 Unmounting a datastore

- Once the datastore has been successfully unmounted, select **Detach** on the disk device.



Name	Type	Capacity	Operational State	Hardware Acceleration	Drive Type	Transport
Local USB Direct-Access (mpx.vmhba32:...	disk	0.00 B	Attached	Not supported	HDD	Block Adapter
COMPELNT Fibre Channel Disk (naa.600...	disk	1.00 TB	Attached	Supported	HDD	Fibre Channel
Local ATA Disk (t10.ATA_____MZ2D5EA1...	disk	93.16 GB	Attached	Unknown	Flash	Block Adapter
Local USB CD-ROM (mpx.vmhba32:C0:T0...	cdrom		Attached	Not supported	HDD	Block Adapter

Figure 20 Detaching a storage device

- Repeat step 1 through step 3 for each host the volume is presented.
- Within the DSM Client, unmap the volume.
- Within the vSphere client, rescan the adapters to ensure that the disk has been removed.

Note: Graceful removal of volumes from an ESXi host is done automatically when using the Dell SC Series vSphere client plug-in. More information about how to obtain the plug-in is in appendix B.

6.14 Mapping volumes from multiple arrays

When mapping volumes from separate arrays to the same ESXi cluster, attention should be paid to the LUN when mapping volumes from each system. For administrative simplicity, it is recommended to use separate numbering schemes when mapping volumes from each of the arrays. For example, array A might map volumes using LUNs 1 through 50, while array B might use LUNs 51 through 100. A numbering scheme helps avoid conflicts if a volume is ever promoted to a Live Volume between the two arrays in the future.

6.15 Multipathing resources

For more information about multipathing, see the section, “Understanding Multipathing and Failover” in the *vSphere Storage Guide* at [VMware vSphere documentation](#).

7 Boot from SAN

Booting ESXi hosts from SAN yields both advantages and disadvantages. Sometimes, such as with blade servers that do not have internal disk drives, booting from SAN may be the only option. However, many ESXi hosts can have internal mirrored drives, providing the flexibility of choice. The benefits of booting from SAN are obvious—it alleviates the need for internal drives and allows the ability to take snapshots (replays) of the boot volume.

However, there are also benefits to booting from local disks and having the virtual machines on SAN resources. Booting from local disks gives ESXi the advantage of staying online if maintenance needs to be performed on Fibre Channel switches, Ethernet switches, or the array itself. The other clear advantage of booting from local disks is using the VMware iSCSI software initiator instead of iSCSI HBAs or Fibre Channel cards.

The decision to boot from SAN depends on many business-related factors including cost, recoverability, and configuration needs. Dell does not offer a specific recommendation.

7.1 Configuring boot from SAN

When deciding to boot ESXi hosts from SAN, a few best practices need consideration.

When mapping the boot volume to the ESXi host for the initial install, the boot volume should only be mapped down a single path to a single HBA. Once ESXi has been loaded and multipath modules are operating correctly, the second path can be added to the boot volume.

To use the advanced mapping screen in DSM, it must be enabled through the **Preferences** menu in the SC Series settings.

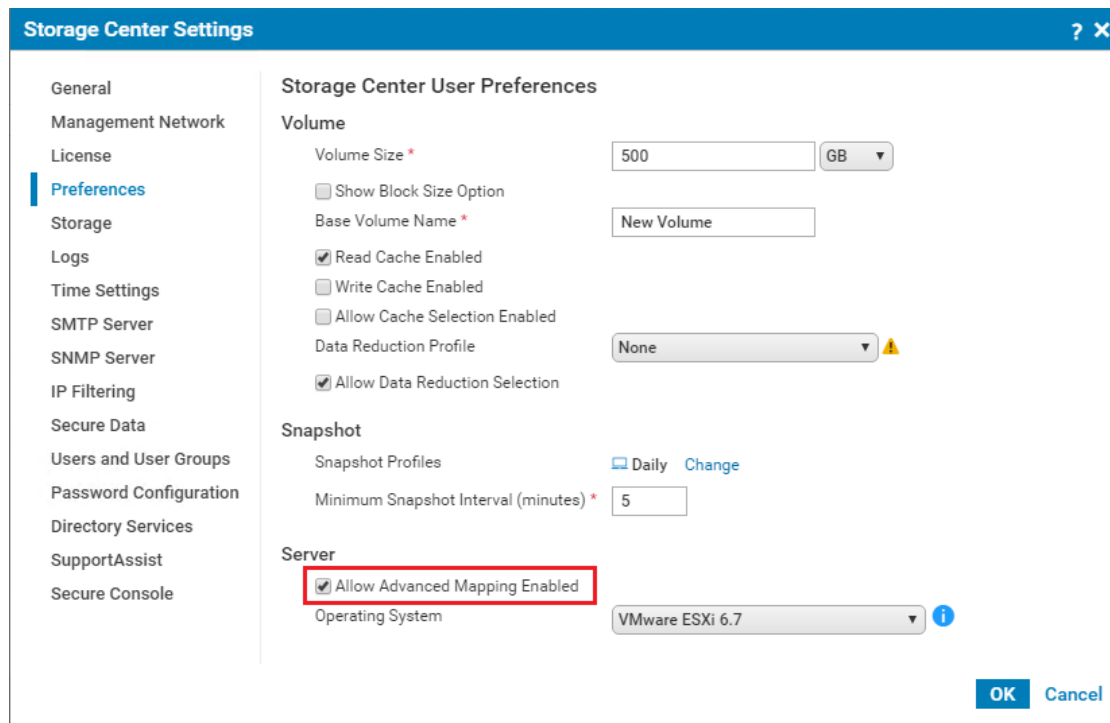


Figure 21 Enabling advanced mapping

Once advanced mapping is enabled, there are a few options that need to be changed for the initial installation, as shown in Figure 22.

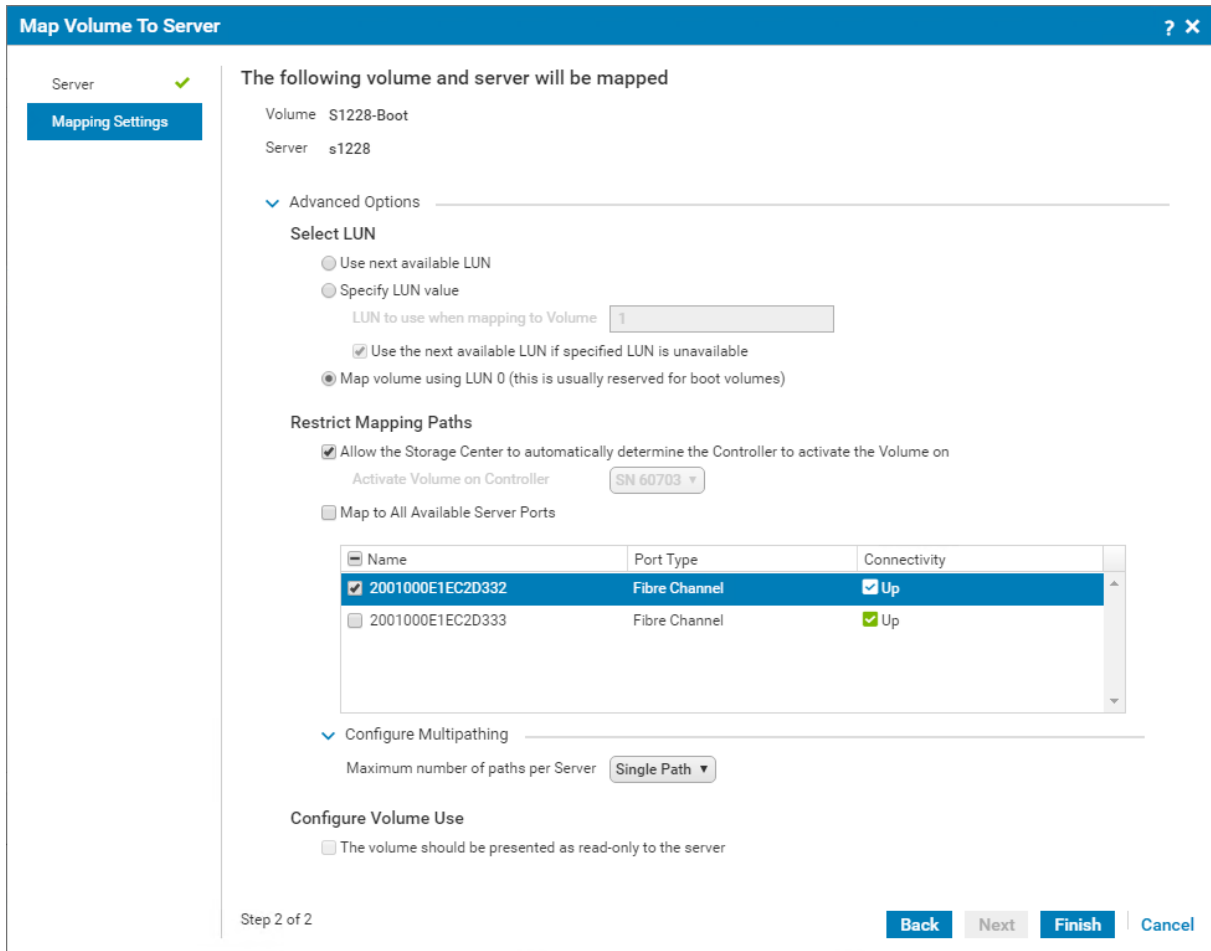


Figure 22 Advanced mapping screen for configuring boot from SAN

Map volume using LUN 0: Checked

Maximum number of paths allowed: Single-path

Once the ESXi host is running correctly, the second path can then be added to the boot volume by modifying the mapping. First, right-click the mapping, select **Modify Mapping**, and select the following:

Maximum number of paths allowed: Unlimited

Once the additional path has been added, the HBAs on the ESXi host can be rescanned.

8 Volume creation and sizing

Administrators are tasked with complex decisions such as determining the best volume size, number of virtual machines per datastore, and file system versions for their environment.

8.1 Volume sizing and the 64 TB limit

Although the maximum size of a volume that can be presented to ESXi is 64 TB, the general recommendation is to start with smaller and more manageable initial datastore sizes and expand them as needed. Remember that a datastore can easily be expanded to a larger size later, so it is prudent to start with datastore sizes in the 500–750 GB range. This is based on the consideration that a 750 GB datastore will accommodate approximately 15 x 40 GB virtual disks. This will leave a small amount of overhead for virtual machine configuration files, logs, snapshots (replays), and memory swap, keeping the datastore performing adequately.

The largest single extent 64 TB VMFS-5 volume size is $(64 \times 1,024 \times 1,024 \times 1,024 \times 1,024 = 70,368,744,177,664)$ bytes. For any volumes larger than this size, VMware will not consume the additional space.

Note: These sizing recommendations are provided to limit the number of virtual machines on each datastore and to keep performance manageable, not for capacity reasons. If there is a virtual machine that requires a large virtual disk, creating a large datastore to accommodate the VM is acceptable.

Note: Within certain versions of ESXi 5.x, hosts may have maximum addressable storage limits of 4 TB, 8 TB, or 60 TB due to the VMFS heap size. Read the following VMware KB article for more information: <http://kb.vmware.com/kb/1004424>.

8.2 Virtual machines per datastore

There are no steadfast rules for the number of virtual machines that should be placed on a datastore. Due to the scalability enhancements of VMFS, a good conservative approach is to place anywhere between 15 to 25 virtual machines on each.

The reason behind limiting virtual machines and VMDK files per datastore is due to potential I/O contention, queue depth contention, or other conflicts that may degrade system performance. For the same reason, it is advisable to limit datastore sizes to 500 to 750 GB. Size helps limit the total number of virtual machines that can be placed on each datastore.

The art to virtual-machine placement revolves around analyzing the typical disk I/O patterns for each of the virtual machines and placing them accordingly. In other words, the sweet spot of how many virtual machines can be put on each datastore is dependent on the disk load. For example, sometimes the appropriate number for high-I/O-load virtual machines may be less than five, while the number of virtual machines with low-I/O disk requirements may be 25 or more.

The appropriate number of virtual machines that can be put onto each datastore is subjective and dependent on the environment. A good recommendation is to start with 15 to 25 virtual machines and increase or decrease the number of virtual machines on each datastore as needed. Moving virtual machines between datastores can even be done nondisruptively when licensed to use the VMware Storage vMotion feature.

The most common indicator a datastore has too many virtual machines is if the queue depth of the datastore is regularly exceeding set limits and increasing disk latency. Remember that if the driver module is set to a queue depth of 256, the maximum queue depth of each datastore is also 256. Meaning that if there are 16 virtual machines on a datastore, all heavily driving a queue depth of 32 ($16 \times 32 = 512$), they are essentially overdriving the disk queues by double. The resulting high latency will most likely degrade performance. See appendix A for more information about determining if the queue depth of a datastore is being correctly used.

In rare situations where (VAAI) hardware-assisted locking is unavailable or disabled, a far less common indicator of overdriving the datastore would be the frequent occurrence of SCSI reservation conflicts. Within esxtop, there is a field in the Disk Device screen for reserve stats (RESVSTATS). When monitoring datastores, it is normal to see a few reservation (RESV/s) entries and even a few conflicts (CONS/s) from time to time. However, when CONS/s happen frequently on a volume, it may be time to move some of the virtual machines to a different datastore. The VAAI hardware-assisted locking primitive will help to alleviate the performance degradation caused by these SCSI-2 reservations. If datastores are impacted, it is recommended to upgrade to a version of SCOS that supports this VAAI primitive. See section 16.3 on VAAI for more information.

Note: Many resources discuss VMware infrastructure design and sizing. The general rule discussed previously may vary based on the needs of the environment.

8.3 VMFS partition alignment

Partition alignment is a performance-tuning technique used with traditional SANs to align the guest operating system partitions and VMFS partitions to the physical spinning media. This alignment reduces the number of disk transactions it takes to process an I/O.

Due to how dynamic block architecture virtualizes the blocks, manual partition alignment is not necessary. SC Series automatically aligns its 512 KB, 2 MB, or 4 MB pages to the physical sector boundaries of spinning drives. Since the largest percentage of performance gains are seen from aligning the SC Series pages to the physical disks, manually tuning has a minor effect on performance.

As found in Dell EMC internal lab testing, any performance gains achieved by manually aligning partitions are not substantial enough ($\pm 1\%$) to justify the extra effort. However, because all workloads are different, performing testing is recommended to determine the impact of an aligned partition for applications.

To manually align the VMFS block boundaries to the SC Series page boundaries for performance testing, the recommended offset when creating a datastore is 8,192 (4 MB).

Note: Using the SC Series vSphere client plug-in to create datastores will automatically align them to the recommended offset. More information about how to obtain the plug-in is in appendix B.

Figure 23 shows an example of a fully aligned partition in the SC Series where one guest I/O will only access necessary disk sectors.

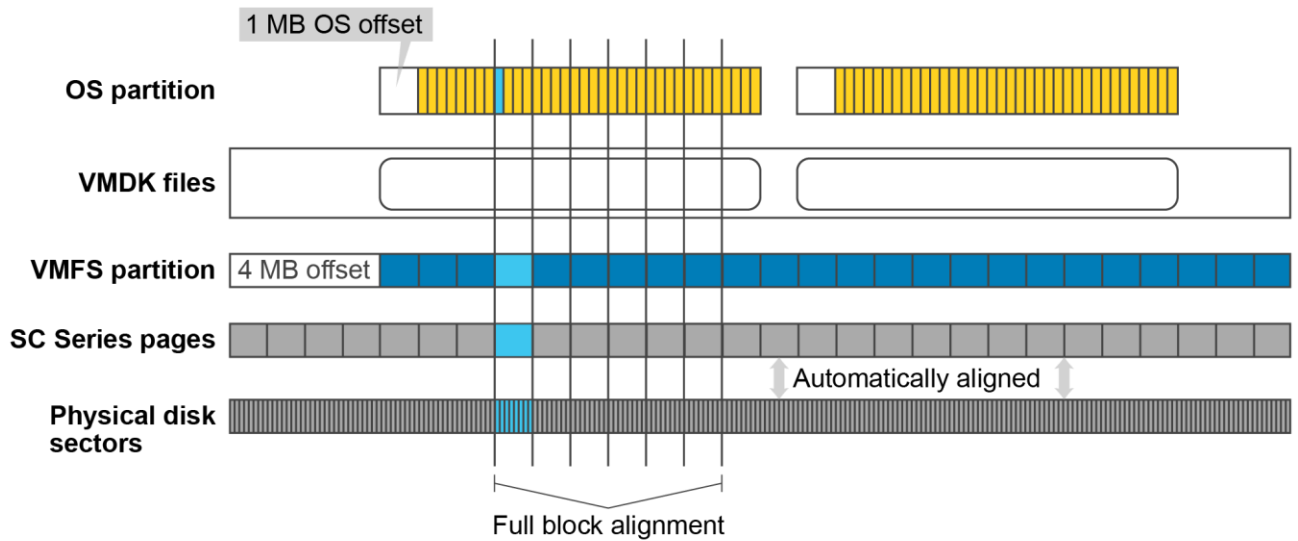


Figure 23 Fully aligned partition in SC Series storage

Figure 24 shows an example of an unaligned partition in a traditional SAN where alignment can improve performance.

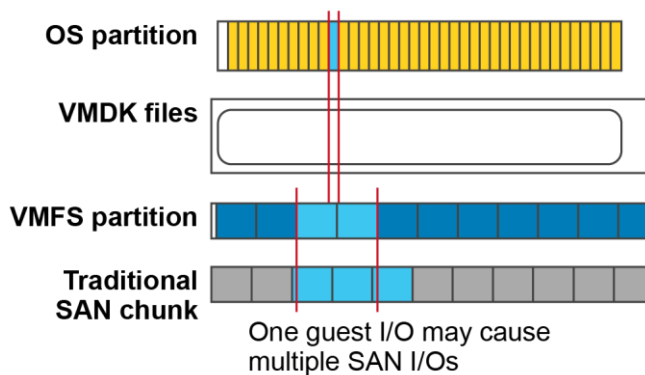


Figure 24 Unaligned partition in a traditional SAN

8.4 VMFS file systems and block sizes

Within ESXi, it is recommended to use VMFS-6, which is the recommended file system format selectable from the vSphere client when creating a datastore.

8.4.1 VMFS-3

If there are any remaining VMFS-3 datastores in the environment, it is recommended they be retired, and virtual machines be migrated to the latest-version VMFS-5/6 formatted datastore. As a best practice, create new VMFS-5/6 datastores, and migrate virtual machines to them using Storage vMotion when possible.

Caution: Before upgrading a VMFS-3 datastore to VMFS-5, it is recommended that a snapshot is taken of the datastore for protection against any possible loss of data.

8.4.2 VMFS-5

With VMFS-5, the default block size is 1 MB, and allows for up to a 64 TB datastore with up to a 62 TB VMDK. This format is required for functionality such as the VAAI space reclamation primitive (SCSI UNMAP) to reclaim storage after a VMDK is deleted. See section 16.3 for more information about VAAI.

8.4.3 VMFS-6

VMFS-6 shares many of the same configuration maximums as VMFS-5, however VMFS-6 allows for automatic space reclamation (UNMAP) for thin provisioned volumes. Since there is no upgrade path from VMFS-5 to VMFS-6, migration techniques such as Storage vMotion must be used to take advantage of the new features.

9 Volume mapping layout

In addition to volume sizing, another important factor to consider is the placement of files and virtual machine data.

9.1 Multiple virtual machines per volume

One of the most common techniques in virtualization is to place more than one virtual machine on each volume. Encapsulation of virtual machines within datastores results in higher consolidation ratios. When deciding how to lay out the VMFS volumes and virtual disks, it should reflect the performance and backup needs of the guest operating systems. Regardless of the layout of the virtual machines, there are some basic concepts that should be considered.

9.1.1 Storage of non-virtual machine files

One general recommendation is to create a content library or a datastore for administrative items to store virtual machine templates, ISO images, virtual floppies, or scripts.

9.1.2 Separation of operating system page files

In situations where memory swapping cannot be avoided, one technique to consider with virtual machine placement is separating the operating system paging files or swap files onto a separate datastore.

There are two main reasons for separating operating system page files onto their own datastore. First, because page files can generate a lot of disk activity, this practice could keep volume snapshots (replays) smaller. Second, if replicating those volumes, this practice will conserve bandwidth by not replicating the operating system page file data.

Depending on the memory-swap conditions unique to each environment, separating page files may or may not make a significant reduction in snapshot sizes. A good way to determine whether separating page files will make a difference is to use the vSphere client performance charts to monitor swap or balloon usage of the ESXi host. If these numbers are high, consider testing the separation of page files to determine the impact.

If separating page files will make an impact in reducing snapshot sizes, the general recommendation is to create pairs of volumes for each datastore containing virtual machines. If a volume is created to contain 10 virtual machines, then a second volume should be created to store the operating system page files for those 10 machines. For example:

Create one datastore for virtual machines: This datastore will usually contain the virtual machine disk (VMDK) files, configuration files, and logs for the virtual machines.

Create one paired datastore for the corresponding virtual machine page files: This datastore should contain virtual machine page files. Using Windows as an example, create a virtual disk (P:) on this volume large enough to store the Windows paging file for each virtual machine. This volume can be sized considerably smaller than the main datastore since it only needs enough space to store page files.

A question often asked is: Should all operating system page files be placed on a single datastore? This practice is not a good idea for the following reasons.

First, the page file datastore can also experience contention from queue depth utilization or disk I/O. Too many VMDK files during a sudden memory-swapping event could decrease performance even further. For example, if a node in the ESXi HA cluster fails and the affected virtual machines are consolidated on the

remaining hosts. The sudden reduction in overall memory could cause a sudden increase in paging activity that could overload the datastore, causing a storage performance decrease.

Second, that datastore could become a single point of failure. Operating systems are not very tolerant of unexpected disk drive removal. If an administrator were to accidentally unmap the page file volume, the number of virtual machines within the failure domain would be isolated to a subset of the virtual machines.

9.1.3 Separation of virtual machine swap files

Each virtual machine also has a memory swap file in its home directory used when the VMware Tools balloon driver is unable to reclaim enough memory. In other words, the virtual machine swap (VSWP) file is only used as a last resort by the ESXi host to reclaim memory. VMware recommends keeping the VSWP files in the virtual machine home directories. However, if needed, it is also possible to relocate the VSWP file to a dedicated volume. While doing this technique may help to reduce snapshot sizes and preserve replication bandwidth, it should only be done under the guidance of VMware support.

9.1.4 Virtual machine placement

This example technique gives a great deal of flexibility when building out the storage architecture in the environment, while keeping with the basic concepts discussed above. The following example layout meets most virtual infrastructure needs because it adds the flexibility of being able to add RDMs to virtual machines later if needed. The key to this technique is reserving LUN numbers in the middle of the LUN sequence to help better organize the virtual machines.

An example of this technique is as follows:

LUN0 - Boot LUN for ESXi (When booting from SAN)

LUN1 – Templates, ISOs, General Storage

LUN10 – OS, DATA (drives C, D, and E)

LUN11 – Page file (Paired with LUN10) for operating system paging files [If necessary]

LUN12 - LUN19 - Reserved LUNs for virtual machine RDMs for machines in this group

LUN20 – OS, DATA (drives C, D, and E)

LUN21 – Page file (Paired with LUN20) for operating system paging files [If wanted]

LUN22 - LUN29 - Reserved LUNs for virtual machine RDMs for machines in this group

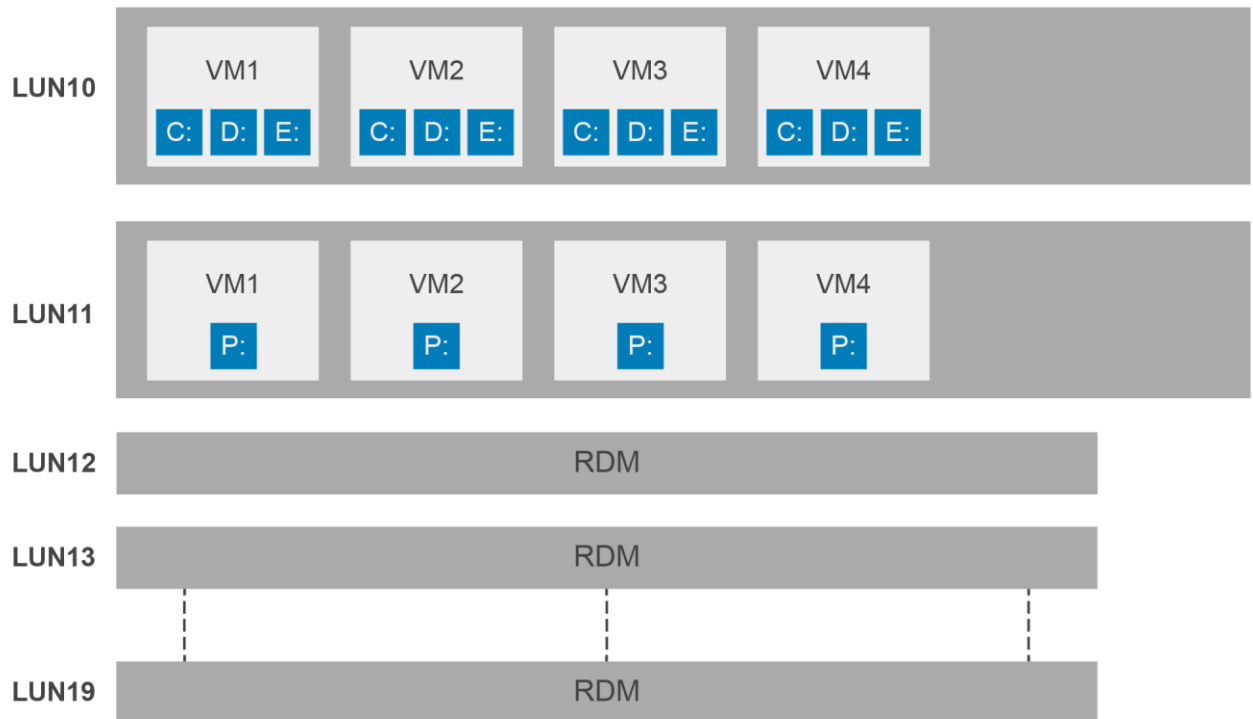


Figure 25 Virtual machine placement (with RDMs)

Timesaver: To help organize the LUN layout for ESXi clusters, some administrators prefer to store their layout in a spreadsheet. Not only does this help to design their LUN layout in advance, but it also improves organization as the clusters grow larger.

Note: Many factors may influence architecting storage regarding the placement of virtual machines. The method shown above is merely a suggestion; business needs may dictate different alternatives.

9.2 One virtual machine per volume

Although creating one volume for each virtual machine is not a common technique, there are both advantages and disadvantages that will be discussed below. Keep in mind that deciding to use this technique should be based on business-related factors and may not be appropriate for all circumstances. Using a 1:1 virtual machine-to-datastore ratio should be the exception, not the rule.

Advantages of creating one volume per virtual machine include:

- Granularity in replication: Since the SC Series replicates at the volume level, if there is one virtual machine per volume, administrators can choose which virtual machine to replicate.
- Reduced I/O contention: A single volume is dedicated to a single virtual machine.
- Flexibility with volume mappings: Since a path can be individually assigned to each volume, it could allow a virtual machine a specific path to a controller.
- Statistical reporting: Storage usage and performance can be monitored for an individual virtual machine.
- Simplified backup and restore of an entire virtual machine: If a VM needs to be restored, an administrator can unmap or remap a snapshot in its place.

Disadvantages of creating one volume per virtual machine include:

- Maximum virtual machines equal to disk device maximums in the ESXi cluster: For example, if the HBA has a maximum limit of 512 volumes that can be mapped to the ESXi host. Since each logical unit number can be used only once when mapping across multiple ESXi hosts. It would have a 512 virtual machine limit (assuming that no extra LUNs would be needed for recoveries).
- Increased administrative overhead: Managing a volume for each virtual machine and all the corresponding mappings may be challenging.

10 Raw device mapping (RDM)

A raw device mapping (RDM) is used to map a volume directly to a virtual machine. When an RDM set to physical compatibility mode is mapped to a virtual machine, the operating system writes directly to the volume bypassing the VMFS file system. There are several distinct advantages and disadvantages to using RDMs, but usually, using the VMFS datastores is recommended instead of using RDMs.

Advantages of using RDMs include:

- Virtual mode RDMs (vRDMs) up to 62 TB in size and physical mode RDMs (pRDMs) up to 64 TB in size can be mapped directly to a guest.
- Before shared VMDKs with vSphere 7.0, a clustered resource (such as Microsoft Cluster Services) could be created with options including:
 - Virtual machine to virtual machine
 - Virtual machine to physical machine
- The volume can be remapped to another physical server for recovery.
- Physical machines can be converted to virtual machines more easily because a physical machine volume can be mapped as an RDM.
- When a VM has special disk performance needs:
 - A slight disk performance increase when using an RDM compared to a VMFS virtual disk due to the lack of contention, no VMFS write penalties, and better queue depth utilization.
 - There can be independent disk queues per RDM.
 - Certain types of SAN software can be used, such as Dell Replay Manager or the Windows free space recovery feature.
- A different storage profile can be assigned to each volume. For example, if a database server has its database and logs separated on to different volumes, each can have a separate storage profile.
- A different snapshot schedule (replay profile) can be used with each volume. For example, a database and its transaction logs may have different snapshot (replay) intervals and retention periods for expiration.
- vSphere snapshots are supported with virtual mode RDMs.

Disadvantages of using RDMs include:

- Added administrative overhead due to the number of mappings.
- There are a limited number of volumes that can be mapped to an ESXi host. If all virtual machines used RDMs for drives, the cluster would have a small maximum number of drives.
- Physical mode RDMs cannot be used with ESXi snapshots. While VMware snapshots are not available for physical mode RDMs, SC Series snapshots can still be used to recover data

Note: All previous RDM-related tasks can be automated using the Dell EMC SC Series vSphere Client Plug-in. More information about how to obtain the plug-in is in appendix B.

11 Data Progression and storage profile selection

Data Progression migrates inactive data to the lower tier of inexpensive storage while keeping the most active data on the highest tier of fast storage, as shown in Figure 26. This works to the advantage of VMware datastores because multiple virtual machines are kept on a single volume.

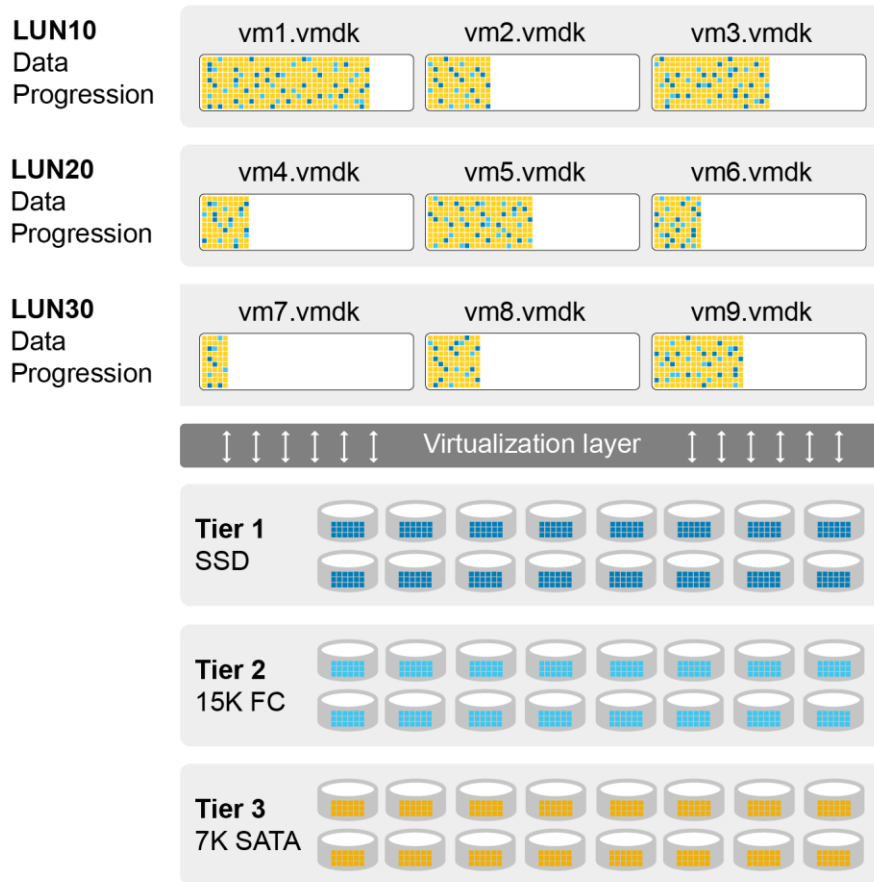


Figure 26 Data Progression and thin provisioning working together

When using Data Progression, virtual machines that have different storage needs, such as different tier or RAID levels, can be placed on the same datastore. This advantage gives the administrator the ability to sort virtual machines by business purpose rather than by disk performance characteristics.

However, if there is a business case where virtual machines would require different RAID types, some decisions on how Data Progression is configured for the volumes must be made.

The following is an advanced example of virtual machine RAID groupings in which forcing volumes into different profiles is wanted.

LUN0 - Boot volume for ESXi

-- Data Progression: Recommended (All Tiers)

LUN1 – Templates, ISOs, General Storage

-- Data Progression: Recommended (All Tiers)

LUN10 – OS, DATA (Server group1 - High performance - four VMs – drives C, D, and E)

-- High Priority (Tier 1)

LUN20 – OS, DATA (Server group2 - Low performance - 15 VMs – drives C, D, and E)

-- Data Progression: Low Priority (Tier 3)

LUN30 – OS, DATA (Server group 3 - Application grouping - five VMs – drives C, D, and E)

-- Data Progression: Recommended (All Tiers)

Unless there is specific business need requiring a virtual machine to be pinned into a specific storage type, tier, or RAID level, it is recommended to keep the configuration simple. Usually, the **Recommended** Data Progression setting can be used to automatically classify and migrate data based on usage.

Note: As a Data Progression best practice, assign a snapshot profile that takes one daily snapshot at a minimum and does not expire for 25 hours or more. This technique will have a dramatic effect on Data Progression behavior and can increase the overall system performance.

11.1 On-Demand Data Progression

With the release of SCOS 6.4 and Dell all-flash arrays, On-Demand Data Progression (ODDP) was introduced. ODDP not only enhances flash performance capabilities, but also adds new tiering possibilities with spinning media. This new capability allows Data Progression to run outside of its normal cycle to move data according to rules defined in the storage profile. For example, with an all-flash system, when a snapshot triggers ODDP, it can migrate data from the write-intensive SSD drives to the read-intensive SSD drives. This will free up space for new incoming writes.

Within a VMware environment, this new functionality may change the storage profile strategies previously implemented in the environment. Administrators should review all the new storage profile selections, such as the ones shown in Figure 27, to see if improvements can be made to their existing tiering strategies.

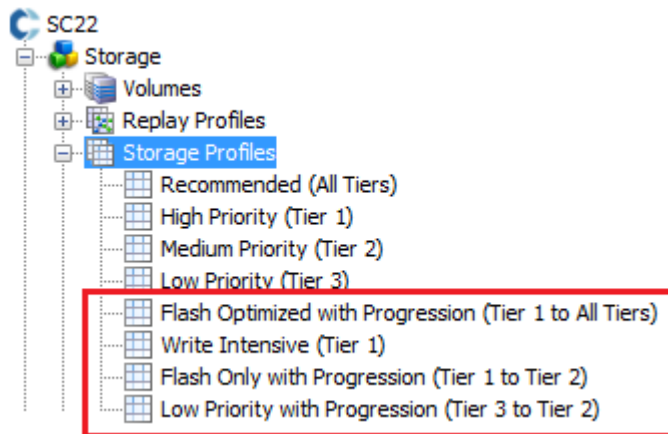


Figure 27 New storage profile available with SCOS 6.4

For detailed information about On-Demand Data Progression, read the *SC Series Storage Manager Administrator's Guide* available on Dell.com/support.

11.2 Data reduction (compression and deduplication)

Released with SCOS 7.0, the data reduction feature takes advantage of both compression and deduplication for reducing the data footprint. As part of its daily cycle, Data Progression deduplicates and compresses the data within volumes that have the data reduction feature enabled. Depending on the data reduction policy set on a particular volume, the data can either be compressed, or deduplicated and then compressed for maximum savings.

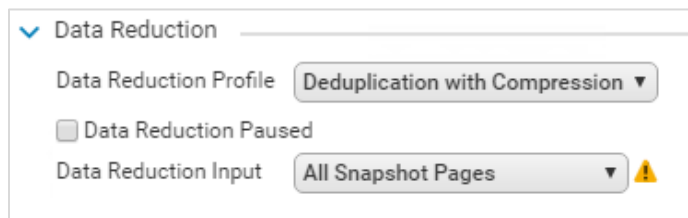


Figure 28 Data reduction options within volume settings

Note: Depending on factors such as array hardware and application workloads, enabling data reduction may have a performance tradeoff in exchange for the capacity savings.

The advanced options available in the volume settings offer control over the data reduction input, which consists of pages an administrator deems eligible for data reduction. The choice between inaccessible snapshot pages and all snapshot pages is a tradeoff between maximum savings and maximum performance.

11.2.1 Data Reduction Input

Within the advanced volume settings, the **Data Reduction Input** setting can be specified to control which pages are eligible for data reduction.

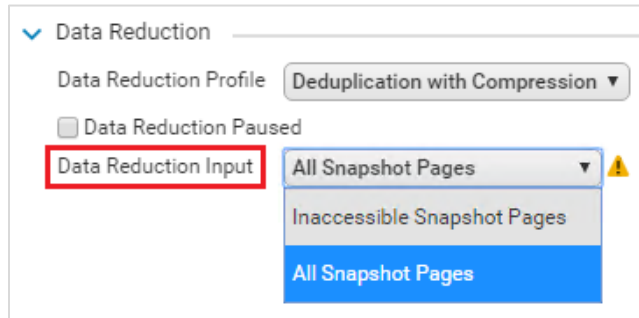


Figure 29 Advanced volume settings showing the Data Reduction Input selection

All Snapshot Pages: All frozen pages in the lowest tier of the system that are part of a snapshot, are eligible for data reduction. If using a single-tiered system, such as a replication target system, all frozen pages are eligible. This setting yields the highest reductions, however there is a performance tradeoff of higher latency due to any accessed pages being decompressed as they are accessed.

Inaccessible Snapshot Pages: Only frozen inaccessible pages are eligible for data reduction. These pages are kept only as part of a snapshot for recovery purposes and are not accessible by the host. This setting yields the lowest reductions however it has the best performance since no host-accessible pages are compressed or deduplicated.

Caution: When initially enabling compression on a system, the Data Progression total running time may be extended and could result in the cycle continuing into regular business hours. Administrators should adjust the Data Progression settings within the system settings to limit the maximum run time, or limit the number of volumes to be compressed each day.

For more information about the data reduction feature, read the white paper, [Dell Storage Center OS 7.0 Data Reduction with Deduplication and Compression](#).

12 Thin provisioning and virtual disks

Dell SC Series thin provisioning allows less storage to be consumed for virtual machines, saving upfront storage costs. This section describes the relationship that this feature has with virtual machine storage.

12.1 Virtual disk formats

In ESXi, VMFS can store virtual disks using one of the four different formats described in the following sections.

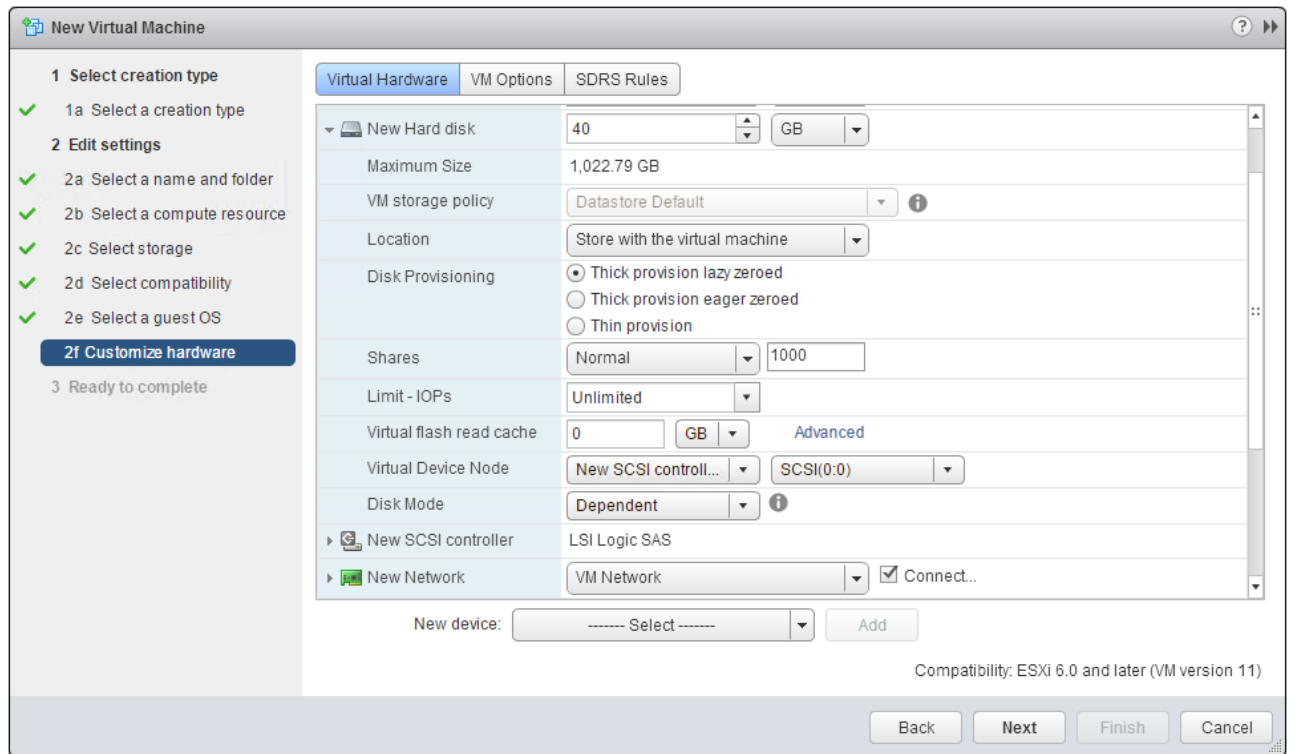


Figure 30 Virtual disk format selection

12.1.1 Thick provision lazy zeroed (zeroedthick)

Only a small amount of disk space is used within the SC Series at virtual disk creation time, and new blocks are only allocated during write operations. However, before any new data is written to the virtual disk, ESXi will first zero out the block to ensure the secure integrity of the write. This on-first-write style of block zeroing before the write induces extra I/O and write latency which could potentially affect applications sensitive to disk latency or performance.

12.1.2 Thick provision eager zeroed (eagerzeroedthick)

Space required for the virtual disk is fully allocated at creation time. Unlike the zeroedthick format, all the data blocks within the virtual disk are zeroed out during creation. Disks in this format may take longer to create than other types of disks because all blocks must be zeroed out before the disks can be used. When using VAAI, the time it takes to create an eagerzeroedthick disk is greatly reduced. This format is generally used for Microsoft clusters and the highest-I/O-workload virtual machines because it does not suffer from operational write penalties like the zeroedthick or thin formats.

12.1.3 Thin provisioned (thin)

The logical space required for the virtual disk is not allocated during creation, but it is allocated on demand during the first write issued to the block. Like thick disks, this format will also zero out the block before writing data, inducing extra I/O, and an additional amount of write latency.

12.1.4 Space efficient sparse (SE sparse disks)

SE sparse disks are a new virtual disk type introduced for use with VMware Horizon View Composer. These virtual disks are used with linked clones for improved performance, space reclamation, and a new 4 KB grain size.

12.2 Thin provisioning relationship

The following points describe how each virtual disk format affects SC Series thin provisioning:

- Thick provision lazy zeroed: SC Series thin provisions the virtual disks.
- Thick provision eager zeroed: SC Series thin provisions the virtual disks (see section 12.3).
- Thin provisioned: SC Series thin provisions the virtual disks. There are no additional storage savings while using this format because the array already uses its thin provisioning (see section 12.4).
- SE sparse (available with VMware Horizon View virtual desktops): SC Series thin provisions the virtual disks. Space reclamation within the VMDK is available.

As a best practice, use the default virtual disk format of thick provision lazy zeroed unless there are specific needs to pre-allocate virtual disk storage. Needs such as Microsoft clustering, VMware Fault Tolerance (FT), or for virtual machines potentially impacted by the thin or zeroedthick on-first-write penalties. If the application is sensitive to VMFS write penalties, it is recommended to test eagerzeroedthick virtual disks to determine the performance impact.

12.3 SC Series thin write functionality

Most versions of SCOS can detect incoming sequential zeroes while being written and track them, but not write the zeroed page to the drives. When creating virtual disks on these versions of firmware, all virtual disk formats will be thin provisioned at the array level, including thick provision eager zeroed.

12.4 SC Series thin provisioning or VMware thin provisioning

A common question is whether to use array-based thin provisioning or the VMware thin provisioned VMDK format. Since SC Series uses thin provisioning on all volumes by default, it is not necessary to use VMware thin provisioning because there are no additional storage savings by doing so.

However, if VMware thin provisioning is needed, pay careful attention not to accidentally overrun the storage allocated. To prevent any unfavorable situations, the integrated vSphere datastore threshold alerting capabilities should be used to warn against running out of space on a datastore. The threshold alerting capabilities of Dell Storage Manager can also be used to alert for low-space conditions.

12.5 Windows free space recovery

Windows NTFS file system with Windows Server® 2012 and prior has a reporting nuance. Over time, the usage of the file system can grow apart from what the array reports as being allocated. In this example, there is a 20 GB data volume where Windows writes 15 GB worth of files, followed by deleting 10 GB worth of

those files. Although Windows reports only 5 GB in-use, thin provisioning has assigned those blocks to that volume, so the array will still report 15 GB of data used. When Windows deletes a file, it merely removes the entry in the file allocation table, and there are no onboard mechanisms for the SC Series to determine if an allocated block is still in use by the operating system. However, the DSM server agent contains the necessary functionality to recover this free space from machines running earlier versions of Windows that do not possess SCSI UNMAP capabilities. It compares the Windows file allocation table to the list of blocks allocated to the volume. Once finished, it then returns those free blocks into the storage pool to be used elsewhere in the system. Blocks kept as part of a snapshot (replay) cannot be freed until that snapshot is expired.

The free space recovery functionality can only be used in Windows virtual machines under the following circumstances:

- The virtual disk needs to be mapped as an RDM set to *physical* compatibility mode (pRDM). The free space recovery agent can then perform a SCSI query of LBAs in use and correlate them to the blocks allocated on the array that can be freed. The disk must be an NTFS basic disk (either MBR or GPT).
- The virtual disk cannot be a VMDK or an RDM set to *virtual* compatibility mode (vRDM).
 - VMware does not provide APIs for the free space recovery agent to correlate the virtual LBAs to the actual physical LBAs needed to perform the space recovery, since UNMAP handles this operation.
 - If a virtual machine has a drive C (VMDK) and a drive D (RDMP), Windows free space recovery will only be able to reclaim space for the drive D.
 - The restriction against using vRDMs for space recovery also implies that these disks cannot participate in ESXi host snapshots. Software that uses VMware snapshots is needed, an alternative method of backing up the pRDMs is needed. For example, the Dell Storage PowerShell SDK installation provides an example PowerShell script. This script can be used to back up physical mode RDMs as part of the pre-execution steps of the backup job.
- The free space recovery agent also works with volumes mapped directly to the virtual machine through the Microsoft software iSCSI initiator. Volumes mapped to the virtual machine through the Microsoft iSCSI initiator interact with the SAN directly, and therefore, space recovery works as intended.

For more information about Windows free space recovery and compatible versions of Windows, consult the *Dell Storage Manager Administrator's Guide* on Dell.com/support.

12.6 Affinity Manager 2.0

With the release of vSphere 7.0, VMware has improved the first-write process when using thin provisioned VMDKs. While VMware still recommends using Eager Zeroed Thick (EZT) virtual disks for maximum performance, this new method should decrease the on-first-write performance overhead previously experienced with thin VMDKs. For an overview of the new feature, see the [VMware vSphere 7 Core Storage whitepaper](#).

13 Extending VMware volumes

Within an ESXi host, there are three ways to extend or grow storage. This section provides the general steps required. Additional information can be found in the *vSphere Storage Guide*, section “Increasing VMFS Datastore Capacity”, and in the *vSphere Virtual Machine Administration Guide*, section “Virtual Disk Configuration” located within the [VMware vSphere documentation](#).

13.1 Increasing the size of VMFS datastores

Two methods can be used to increase the size of VMFS datastores: Expanding an existing extent (recommended) or adding a new extent (not recommended).

13.1.1 Expanding an extent in an existing VMFS datastore

This functionality is used to grow an existing extent in a VMFS datastore, but it can be done only if there is adjacent free capacity.

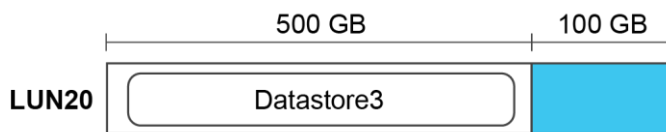


Figure 31 Growing Datastore3 using the adjacent 100 GB (in blue)

In Figure 31, extending the space at the end of an SC Series volume can be done using Dell Storage Manager or the vSphere client plug-ins. After the volume has been extended and HBA rescanned, click **Increase...** to edit the properties of the datastore to grow and follow the Increase Datastore Capacity wizard instructions.

Be sure to select the **Expandable** volume, otherwise a VMFS extent will be added to the datastore (see section 13.1.2 on VMFS extents).

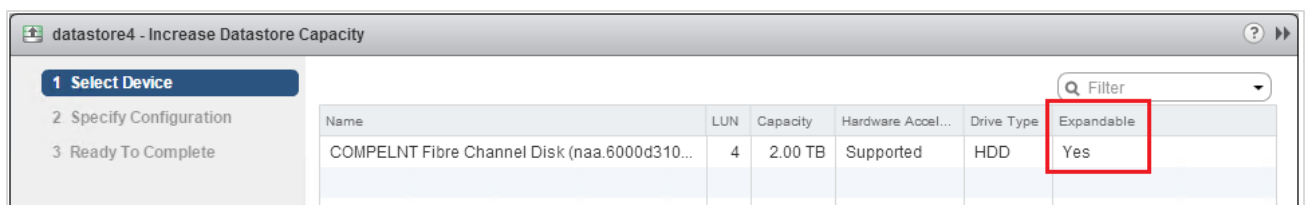


Figure 32 Extending a 2 TB datastore by 1 TB

Note: As an alternative to extending a datastore volume when a virtual machine needs additional disk space, create a datastore volume and migrate that virtual machine. This technique helps to keep volume sizes manageable and helps to keep any single datastore from being overloaded due to I/O contention.

Note: All prior tasks in this section can be automated by using the Dell SC Series vSphere Client Plug-in. More information about how to obtain the plug-in is in appendix B.

13.1.2 Adding a new extent to an existing datastore

This legacy functionality was used with previous versions of vSphere to concatenate multiple volumes to create VMFS-3 datastores larger than 2 TB. Since the maximum datastore size with VMFS-5 has been increased to 64 TB, the use of extents is no longer necessary.

Caution: Due to the complexities of coordinating snapshots (replays) and recoveries of datastores that are spanned across multiple SC Series volumes, the use of VMFS extents is highly discouraged. However, if the use of extents is necessary, snapshots of those volumes should be taken using the consistent snapshot schedule (replay profile) functionality.

13.2 Increasing the size of a virtual machine disk (VMDK) file

Hot extending a SCSI virtual disk is available from within the vSphere client when editing the settings of a virtual machine.

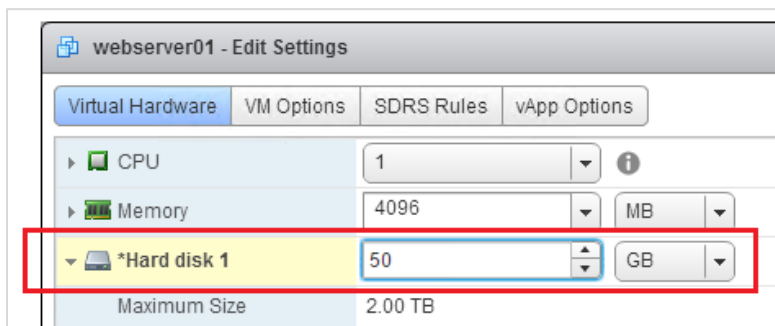


Figure 33 Growing a virtual disk from the virtual machine properties screen

After growing the virtual disk from the vSphere client, an administrator must connect to the virtual machine, rescan for new disks, and then extend the file system.

Caution: Microsoft does not support extending the system partition (drive C) of a machine in certain versions of Windows.

13.3 Increasing the size of a raw device mapping (RDM)

To extend a raw device mapping, follow the same basic procedure as with a physical server. First, extend the RDM volume from within the DSM Client, rescan disks from Windows disk management, and then use DISKPART or the Storage MMC console to extend the drive.

A physical mode RDM whose placeholder VMDK resides on a VMFS-5 datastore can be extended up to the 64 TB limit. However, depending on the version, vRDMs may have a limit anywhere between 2 TB to 62 TB.

Caution: Like datastore volumes, it is also important not to extend an RDM volume with its pointer file residing on a VMFS-3 datastore past the 2047 GB (1.99 TB) limit.

14 Snapshots (replays) and virtual machine backups

Backup and recovery are important to any virtualized infrastructure. This section discusses several common techniques to improve the robustness of virtualized environments such as using snapshots (replays) and virtual machine backups.

14.1 Backing up virtual machines

The key to any good backup strategy is to test the backup and verify the results. There are many ways to back up virtual machines, but depending on business needs, each solution is unique to each environment. Testing and verification may prove that one solution works better than another does, so it is best to test a few different options.

Since the subject of virtual machine backup is vast, this section only covers a few basics. For more information about virtual machine backup strategies, consult VMware documentation.

14.1.1 Backing up virtual machines to tape or disk

Perhaps the most common methods of backing up virtual machines to tape are using third-party backup software or using the backup client software installed within the guest. Options include:

Backup software using vStorage APIs for Data Protection (VADP): VADP is a successor to VMware Consolidated Backup and provides an integrated method to back up virtual machines for backup solutions such as Dell NetVault™ Backup for VMware. For more information, see [VMware vSphere Storage APIs - Data Protection \(VADP\) FAQ](#) in the VMware Knowledge Base.

Backup client loaded within the guest: Using this method, traditional backup software is loaded within the guest operating system, and the data is backed up over the network to a backup host containing the tape drive. Depending on the software used, it only performs file-level backups, but sometimes, it can include additional capabilities for application-level backups.

14.1.2 Backing up virtual machines using snapshots

The options for backing up virtual machines using SC Series snapshots are described as follows:

Snapshots scheduled from within the DSM client: From within the DSM client, a snapshot schedule (replay profile) can be created to schedule snapshots of virtual machine volumes. Usually, using snapshots to back up virtual machines is enough to perform a standard recovery. Remember, snapshots can only capture data that has been written to disk, and therefore the virtual machine data is preserved in what is called a crash-consistent state. In other words, when recovering the virtual machine, the data is recovered as if the virtual machine had lost power. Most modern journaling file systems such as NTFS or EXT3 are designed to recover from such states.

Snapshots taken using Dell Replay Manager software: Since virtual machines running transactional databases are more sensitive to crash-consistent data, Replay Manager utilizes the Microsoft VSS framework for taking snapshots of VSS-aware applications. This software package will ensure that the application data is in a consistent state before performing the snapshot.

Snapshots taken using Dell scripting tools: For applications that need a custom method for taking consistent snapshots of the data, Dell has developed two scripting tools:

- Dell Storage PowerShell SDK: This scripting tool also allows scripting for many of the same storage tasks using the Microsoft PowerShell scripting language.
- Dell Compellent Command Utility (CompCU): A Java-based scripting tool that allows scripting for many of the SC Series tasks (such as taking snapshots).

A good example of using one of these scripting utilities is writing a script to take a snapshot of an Oracle database after it is put into hot backup mode.

Snapshots used for SC Series replication and VMware Site Recovery Manager: Replicating snapshots to a disaster recovery site ensures an offsite backup. In addition, when using Site Recovery Manager, it provides an automated recovery of the virtual infrastructure in the event a disaster is declared.

14.2 Recovering virtual machine data from a snapshot

When recovering a VMFS datastore from a snapshot (replay), an administrator can recover an entire virtual machine, an individual virtual disk, or files within a virtual disk.

The basic steps are as follows:

1. Within the DSM client, select the snapshot to recover from and select **Create Volume From Replay**.
2. Continue through the local recovery wizard to create the view volume and map it to the ESXi host designated to recover the data. Be sure to map the recovery view volume using a LUN that is not already in use.
3. Select a host or cluster, select **Storage**, and select **Rescan Storage** to detect the new volumes.
4. With the host or cluster still selected, select **Storage**, select **New Datastore** to open the wizard, and take the following steps:
 - a. Location: Verify the cluster or host and click **Next**.
 - b. Type: Select **VMFS** and click **Next**.
 - c. Datastore name: Enter a name for the recovery datastore.
 - i. Select a host that has visibility to the new volume.
 - ii. Select the LUN for the view volume that was mapped to the host and click **Next**.
 - d. Mount option: Select from three options:
 - > Assign a New Signature: This option regenerates the datastore allowing access by the host (select this option if you are unsure of which option to use).
 - > Keep the Existing Signature: This option should only be used if the original datastore is not present on the host.
 - > Format the disk: This option formats the view volume and creates a datastore from it.
 - e. Ready to complete: Verify all selections and select **Finish**.

Once the datastore has been resignedatured, the snap datastore is accessible.

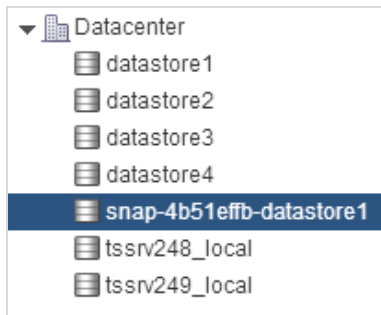


Figure 34 The datastores tab showing the snapshot datastore

The recovery datastore is now designated with **snap-xxxxxxx-originalname**. From this tab, the datastore can be browsed to perform the recovery using one of the methods listed below.

Note: All prior tasks in this section can be automated by using the recovery functionality in the SC Series vSphere Client Plug-ins. More information about how to obtain the plug-in is in appendix B.

14.2.1 Recovering a file from a virtual disk

To recover a file from within a virtual disk on this snap datastore:

1. Select **Edit Settings** for the virtual machine.
2. Choose **Add an Existing Hard Disk**.
3. Select **Use an Existing Virtual Disk**.
4. Browse to select the virtual disk to recover from and browse the snap datastore for the virtual disk containing the data to be recovered.
5. From within the operating system, assign a drive letter to the virtual disk and recover, copy, or move the file back to its original location.
6. After completing the file recovery, remove the recovered virtual disk from the virtual machine before unmapping or deleting the view volume.

14.2.2 Recovering an entire virtual disk

To recover an entire virtual disk from the snap datastore:

1. Browse to the virtual disk to be recovered.
2. Right-click the virtual disk and select **Move to**.
3. Using the wizard, browse to the destination datastore and folder, and click OK.

If a VMDK file is being moved back to its original location, remember that the virtual machine must be powered off to overwrite the virtual disk. Depending on the size of the virtual disk, this operation may take anywhere between several minutes to several hours to finish.

Note: Alternatively, the old VMDK can be removed from the virtual machine. The recovered virtual disk can be re-added to the virtual machine. Then Storage vMotion can be used to move the virtual disk back to the original datastore while the VM is powered on.

14.2.3 Recovering an entire virtual machine

To recover an entire virtual machine from the snap datastore:

1. Browse to the virtual machine configuration file (*.vmx).
2. Right-click the file and select **Register VM**.
3. Use the wizard to add the virtual machine into inventory.

Caution: To prevent name or IP address conflicts when powering on the newly recovered virtual machine, power off or use an isolated network or private vSwitch.

Note: Once the virtual machine has been recovered, it can be migrated back to the original datastore using Storage vMotion.

15 Replication and remote recovery

SC Series replication in coordination with the vSphere line of products can provide a robust disaster recovery solution. Because each replication method affects recovery differently, choosing the correct method to meet business requirements is important. This section provides a brief summary of the different options.

15.1 Synchronous replication

In a synchronous replication, the data is replicated in real time to the destination, and an I/O must be committed on both systems before an acknowledgment is sent back to the host. Sync replication limits the type of links that can be used because they need to be highly available with low latencies. High latencies across the link will slow down access times on the source volume.

For SCOS versions before 6.3, the downside of synchronous replication was that snapshots on the source volume were not replicated to the destination. Any disruption to the link would force the entire volume to be re-replicated from scratch. In versions 6.3 and later, the synchronous replication engine was re-written to remedy these limitations.

In SCOS 6.3 and later, in addition to snapshots being replicated, two synchronous replication modes were introduced. High availability and high consistency modes control how the source volume behaves when the destination volume becomes unavailable.

High availability mode: Accepts writes and journals them to the source volume when the destination volume is unavailable (or when latency is too high) to avoid interrupting service. However, if writes are accepted to the source volume, the destination volume data becomes stale.

High consistency mode: Prevents writes to the source volume when the destination volume is unavailable to guarantee that the volumes remain identical. However, the source volume cannot be modified during this time, which can interrupt operations.

Keep in mind that synchronous replication does not make both the source and destination volumes writeable. That functionality is inherent within the SC Series Live Volume feature.

15.2 Asynchronous replication

In an asynchronous replication, the I/O needs only to be committed and acknowledged to the source system, and the data can be transferred to the destination in a nonconcurrent timeframe. There are two different methods to determine when data is transferred to the destination:

Frozen snapshot: The snapshot schedule dictates how often data is sent to the destination. When each snapshot is taken, the SC Series array determines which blocks have changed since the last snapshot (the delta changes), and then transfers them to the destination. Depending on the rate of change and the bandwidth, it is entirely possible for the replications to fall behind. Monitor replications to verify that the recovery point objective (RPO) can be met.

Replicating the active snapshot: With this method, the data is transferred in near real-time to the destination, potentially requiring more bandwidth than if the system were only replicating the snapshots. As each block of data is written on the source volume, it is committed, acknowledged to the host, and then transferred to the destination as fast as it can. Keep in mind that the replications can still fall behind if the rate of change exceeds available bandwidth.

Asynchronous replications usually have more flexible bandwidth requirements, making it the most common replication method. Another benefit of asynchronous replication is that snapshots are transferred to the destination volume, allowing for checkpoints at the source system and destination system.

15.3 Replication considerations with standard replications

One key consideration about SC Series replication is that when a volume is replicated either synchronously or asynchronously, the replication only flows in one direction. Changes made to the destination volume are not replicated back to the source. Do not to map the replication destination volume directly to a host, but instead to create a read-writable view volume.

Block changes are not replicated bidirectionally with standard replication. The ability to vMotion virtual machines between source controllers (main site), and destination controllers (DR site), is not possible with a standard replication, but is possible with Live Volume.

There are some best practices for replication and remote recovery to consider. ESXi host hardware is needed at the DR site in which to map replicated volumes in the event the source ESXi cluster becomes inoperable. Also, preparations should be made to replicate all management resources to the DR site, including vCenter, DSM, and other management software hosts.

15.4 Replication considerations with Live Volumes

Live Volume is a feature that allows a volume to be accessed from two disparate SC Series systems, allowing for the migration of data, increased uptime, and planned maintenance. Although this technology enables new functionality such as long-distance vMotion, there are some caveats described in the following sections. Administrators need to be aware of these caveats when choosing to use asynchronous or synchronous Live Volume.

Caution: LUN conflicts should be considered when designing Live Volumes into the environment's architecture. When a Live Volume is created, it attempts to keep LUN assignments identical between sites. For example, if a VMFS volume is mapped as LUN 10 at the primary site, it attempts to map the destination Live Volume as LUN 10. In some instances, the secondary site may already have existing LUNs that conflict and cause the Live Volume to not operate as expected. For example, having a Live Volume presented using LUN 10 from the first array, and presented as LUN 11 from a second array will likely cause problems. In addition, this is unsupported with certain configurations such as with vMSC clusters.

15.4.1 Asynchronous Live Volume

When a Live Volume replication is created, the volume becomes read-writable from both the main system and secondary system, allowing vMotion of virtual machines over distance. However, the VMware long-distance vMotion best practices need to be followed. The vMotion network between ESXi hosts must be gigabit or greater, round-trip latency must be 10 milliseconds or less (dependent on vSphere support agreement level), and the virtual machine IP networks Layer 2 must be stretched between data centers. In addition, the storage replication must also be high bandwidth and low latency to ensure Live Volumes can be kept synchronized. The amount of bandwidth required to keep Live Volumes synchronized highly depends on the environment, so testing is recommended to determine the bandwidth requirements for the implementation.

Due to the asynchronous nature of the replication and how it proxies data, it is recommended that asynchronous (async) Live Volumes only are used for disaster avoidance or planned migration. Live Volume is most practically used for planned maintenance operations such as migrating workloads between racks or

data centers. Because of the nature of the proxied I/O, any disruption to the link or primary Live Volume causes the secondary Live Volume datastore to become unavailable as well. If for any reason the primary Live Volume goes down permanently, the administrators need to perform a recovery on the secondary Live Volume from the last known good snapshot. The DSM replication disaster recovery wizard is designed to help with this type of recovery.

15.4.2 Synchronous Live Volume

SCOS 6.5 introduced synchronous capabilities to Live Volume using the newly revamped sync replication engine added in a previous release. Synchronous (sync) Live Volume is a new disaster recovery option to save time and effort recovering virtual machines. Since the disk signatures of the datastores remain the same between sites, volumes do not need to be resignatured, nor do virtual machines need to be re-added to inventory. When using sync Live Volume without automatic failover enabled, the disaster must still be declared through DSM to bring the secondary Live Volumes online. Saving most of the laborious recovery steps previously required, and virtual machines can be powered on after the volume comes online and the host has been rescanned.

One key consideration to using sync Live Volume is that round-trip link latency is an important factor for application performance. In relation, the high consistency mode is the default option to ensure data integrity, meaning that low latencies are especially important. Sync Live Volume is most practically used within a data center or a campus environment where round-trip link latency can be kept low. Although round-trip latencies between sites have no hard-set limitation, it is limited by the tolerances for each application or VMware support tolerances as noted previously.

For more information about Live Volume, consult the paper, [Dell EMC SC Series Storage: Synchronous Replication and Live Volume](#).

15.4.3 Live Volume automatic failover

When using SCOS 6.7 or later, Live Volume automatic failover may be used for stretched cluster configurations such as the vSphere Metro Storage Cluster (vMSC) solution certified by VMware. When using DSM as a tiebreaker at an independent third site, it allows multiple SC Series arrays to survive an entire array failure for increased storage availability scenarios.

For more information about the Live Volume vMSC solution, consult the VMware Knowledge Base article, ["Implementing vSphere Metro Storage Cluster using Dell Storage Live Volume \(2144158\)"](#), and the Dell paper [Dell EMC SC Series Storage: Synchronous Replication and Live Volume](#).

15.5 Replication tips and tricks

Since replicated volumes often contain more than one virtual machine, it is recommended that virtual machines are sorted into specific replicated and non-replicated volumes. For example, if there are 100 virtual machines in the ESXi cluster, and only eight of them need to be replicated to the DR site. A special replicated volume should be created to place the eight virtual machines.

As mentioned previously, if separation proves to reduce snapshot sizes, operating system page files should be kept on a separate volume that is not replicated. That keeps replication and snapshot sizes smaller because the data in the page file changes frequently and it is usually not needed for a system restore.

To set replication priorities, an administrator can take advantage of the SC Series QoS to prioritize replication bandwidth of certain volumes. For example, if there is a 100 Mb connection between sites, two QoS

definitions can be created. The critical volume could get 80 Mb of the bandwidth, and the lower priority volume could get 20 Mb of the bandwidth.

15.6 Virtual machine recovery at a DR site

When recovering virtual machines at the disaster recovery site, the same general steps as outlined in the section 14.2 should be followed.

Timesaver: If the environment has a significant number of volumes to recover, time can be saved during the recovery process by using the replication recovery functionality within the DSM software. These features allow an administrator to predefine the recovery settings with specifics such as the appropriate hosts, mappings, LUN numbers, and host HBAs. After the recovery has been predefined, a recovery at the secondary site is mostly automated.

Caution: It is important that the destination volume, denoted by **Repl of**, is never directly mapped to an ESXi host while data is being replicated. Doing so will inevitably cause data integrity issues in the destination volume, requiring the entire volume to be re-replicated from scratch. The safest recovery method is to always restore the virtual machine from a local recovery or view volume as shown in previous sections.

16 VMware storage features

The vSphere platform has several features that correspond with SC Series features. This section details considerations that should be made about features such as Storage I/O Controls (SIOC), storage distributed resource scheduler (SDRS), and VAAI.

16.1 Storage I/O Controls (SIOC)

SIOC is a feature that was introduced in ESX/ESXi 4.1 to help VMware administrators regulate storage performance and provide fairness across hosts sharing a volume. Because SC Series storage uses a shared pool of disks which can have differing performance characteristics, practice caution when using this feature. Data Progression migrates portions of volumes into different storage tiers and RAID levels at the block level. This behavior ultimately affects the latency of the volume and could trigger the resource scheduler at inappropriate times. It may not make sense to use SIOC unless pinning volumes into specific tiers of disk.

In one example that illustrates this scenario (shown in Figure 35), a datastore contains multiple virtual disks, and each virtual disk has different portions of blocks in T1 and T3. If VM1 begins to read a large amount of archived data from vm1.vmdk residing on T3 disks, increasing the latency of the datastore above the congestion threshold. The scheduler could activate and throttle vm3.vmdk although most of its blocks reside on a separate tier of disks.

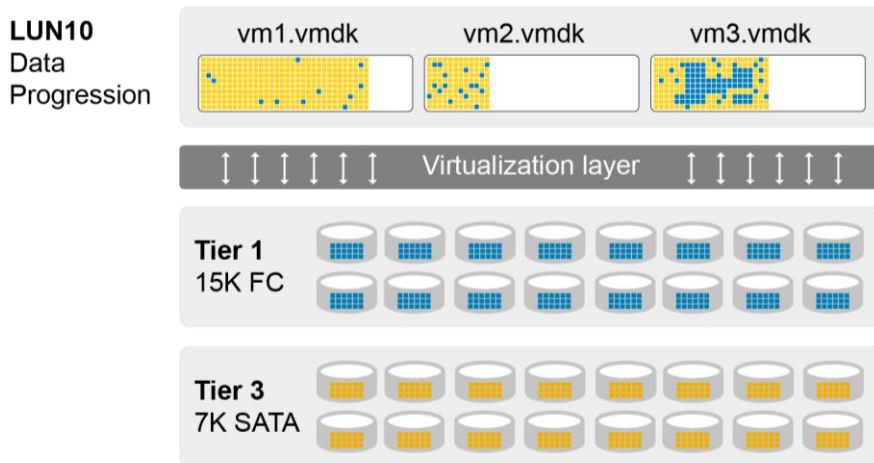


Figure 35 Multiple virtual disks with blocks residing in multiple tiers

As shown in Figure 36, the default setting for the congestion threshold is 30 milliseconds of latency or 90 percent of peak throughput.

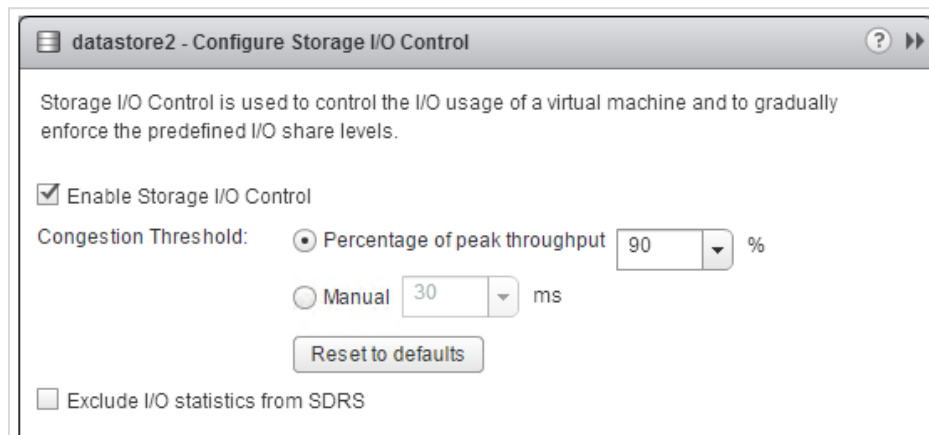


Figure 36 Setting the congestion threshold

As a best practice, leave this setting at default value unless under the guidance of VMware or Dell Support.

For more information about SIOC, refer to the *vSphere Resource Management Guide* at [VMware vSphere documentation](#), or the article, [Unmanaged I/O workload detected on shared datastore running Storage I/O Control \(SIOC\) for congestion management](#) in the VMware Knowledge Base.

16.1.1 SIOC and volume QoS

Similar to VMware SIOC, SCOS 7.0 introduces the volume QoS feature, which allows array administrators to limit bandwidth, IOPS, and prioritize storage workloads at the array level. While SIOC allows vSphere administrators control over storage within the vSphere cluster, volume QoS can balance all array workloads external to vSphere to prevent a noisy neighbor effect. Volume QoS allows all workloads within the array to be balanced for fairness.

Before implementing volume QoS, refer to the guide [Volume Quality of Service Best Practices with Dell SC Series Storage](#).

16.2 Storage distributed resource scheduler (SDRS)

SDRS is a feature introduced in ESXi 5.0 that automatically load balances virtual machines within a datastore cluster based on capacity or performance. When creating SDRS datastore clusters with SC Series storage, remember a few guidelines:

- Group datastores with similar disk characteristics, such as replicated, non-replicated, storage profile, application, or performance.
- Use SDRS for initial placement based on capacity. When placing virtual machines on the datastore cluster, it places them based on which datastore has the most space available.
- Set the automation level to manual mode. SDRS will make recommendations about moving virtual machines, but not move them automatically. As a best practice, run SDRS in manual mode to examine all recommendations before applying them. There are a few items to keep in mind before applying the recommendations:
 - Virtual machines moved between datastores automatically will have their data progression history reset. Depending on the storage profile settings, the entire virtual machine could be moved back to tier 1 RAID 10 if the volume was set to use the recommended storage profile.
 - When a virtual machine moves between datastores, its location at the time the snapshot was taken may make the virtual machine harder to find during a recovery. For example, if a VM moves twice a week while daily snapshots are being taken, the volume with the good snapshot of the virtual machine may be difficult to locate.
 - If the SCOS version in use does not support VAAI, the move process could be slow (without full copy) or could leave storage allocated on its originating datastore (without dead space reclamation). See the section 16.3 on VAAI for more information.
- Storage DRS could produce incorrect recommendations for virtual machine placement when I/O metric inclusion is enabled on an SC Series system using Data Progression. When a datastore is inactive, the SIOC injector performs random read tests to determine latency statistics of the datastore. With Data Progression enabled, the blocks that SIOC reads to determine datastore performance, could potentially reside on SSD, 15 K, or even 7 K drives. This random sampling could ultimately skew the latency results and decrease the effectiveness of the SRDS recommendations.



Figure 37 Disabling I/O metric inclusion

To reiterate, use SDRS only for capacity recommendations, set it to manual automation mode, and disable the I/O metric inclusion. These recommendations will allow administrators to take advantage of SDRS capacity placement recommendations, while still allowing Data Progression to manage the performance of the data at the block level.

16.3 vStorage APIs for array integration (VAAI)

ESXi 5.x introduced five primitives that the ESXi host can use to offload specific storage functionality to the array. These primitives are all enabled by default in vSphere 6.x because they are based on new T10 standardized SCSI-3 commands.

Caution: Using VAAI requires multiple software version prerequisites from both VMware and Dell. Because certain primitives are only available with specific versions of SCOS, consult the [VMware Compatibility Guide](#) for the latest VAAI-certified versions.

16.3.1 Block zeroing (SCSI WRITE SAME)

Traditionally, when an ESXi host creates an eagerzeroedthick virtual disk, it transmits all the zeros over the SAN to the array, which consumes bandwidth, processor, and disk resources. The block zeroing primitive uses the SCSI WRITE SAME command to offload the heavy lifting to the array. For example, when the ESXi host needs a 40 GB VMDK zeroed out, it sends WRITE SAME commands asking the array to write 40 GB's worth of zeros. The array responds to the host when finished. With the SC Series array, since the array does not write zeros due to its thin write feature, creating an eagerzeroedthick virtual disk only takes seconds instead of minutes.

16.3.2 Full copy (SCSI EXTENDED COPY)

The full copy primitive is used to offload to the array the copying or movement of blocks. It does this with the SCSI EXTENDED COPY command that allows the ESXi host to instruct the SC Series array on which blocks it needs copied or moved, leaving the heavy lifting to the array. This has a performance benefit to virtual machine cloning and storage vMotions because the ESXi host no longer must send that data over the wire. The ESXi host tells the SC Series array which blocks need to be copied or moved and the array takes care of the operations.

16.3.3 Hardware accelerated locking

The hardware accelerated locking primitive is intended to add scalability to the VMFS-5 file system by removing the need for SCSI-2 reservations. With VMFS, when a host needs exclusive access to update metadata, traditionally it will set a SCSI-2 nonpersistent reservation on the volume to guarantee it had exclusive rights. Typically, during this operation, VMware documents small performance degradation to other virtual machines accessing this volume simultaneously from different hosts. With the new hardware accelerated locking primitive, it uses a new SCSI-3 method called atomic test and set (ATS) that greatly minimizes any storage performance degradation during the operation. Hardware accelerated locking was added to migrate away from the SCSI-2 operations to SCSI-3 and increase the future scalability of VMFS.

16.3.4 Dead space reclamation (SCSI UNMAP)

The dead space reclamation primitive allows enhanced integration with arrays offering thin provisioning. Before VAAI, when a VMDK was deleted, the array couldn't detect the unused blocks, so they could not be freed back into the pagepool to be reused. This caused a high-water-mark effect in which blocks of data no longer needed by the operating system could still be consumed by the array. The dead space reclamation primitive uses the SCSI UNMAP command. This command allows the ESXi host to instruct the array when specific blocks are no longer in use, and the array can return them to the pool to be reused. If a VMDK is moved or deleted, those blocks will be returned to the pagepool with two notable exceptions. First, if those blocks are frozen as part of a snapshot, they will not return to the pagepool until that snapshot has expired. Second, when using VMFS-5 and earlier, dead space reclamation does not work within the VMDK, and only

works at the VMFS level. If a large file is deleted from within a VMDK, the space would not be returned to the pagepool unless the VMDK itself was deleted.

Note: In most patch levels of ESXi, the dead space reclamation primitive must be invoked manually. See the article, [Using esxcli in vSphere 5.5 and 6.0 to reclaim VMFS deleted blocks on thin-provisioned LUNs](#), in the VMware Knowledge Base.

With vSphere 6.5, the default space reclamation behavior is different compared to other versions. When using VMFS-6 formatted datastores, space reclamation is enabled by default on all datastores, and the reclamation process is automatically invoked by the ESXi hosts. The automatic space-reclamation process operates asynchronously at low priority and is not immediate. In addition, certain pages will not be freed back into the SC Series page pool until after the daily Data Progression cycle has completed.

With vSphere 6.7, the maximum rate at which automatic space reclamation occurs can be increased using the **Space Reclamation Settings** window (see Figure 38). The default rate of **100 MB/sec** is intended to minimize impact on VM I/O. Higher performing arrays, such as Dell EMC SC All-Flash arrays, can operate with a higher automatic space-reclamation rate with minimal impact to VM I/O.

The reclamation rate can be increased if faster reclamation is wanted (for example, with SC All-Flash arrays). However, a higher automatic space-reclamation rate may cause an impact to VM I/O or other I/O served, depending on the configuration and the overall load on the array. When adjusting this value, the recommended starting point is **500 MB/s**, and this setting can be increased or decreased as the load permits.

For SC Series hybrid flash arrays or arrays with all spinning disks, it is recommended not to alter the default rate at which automatic space reclamation occurs.

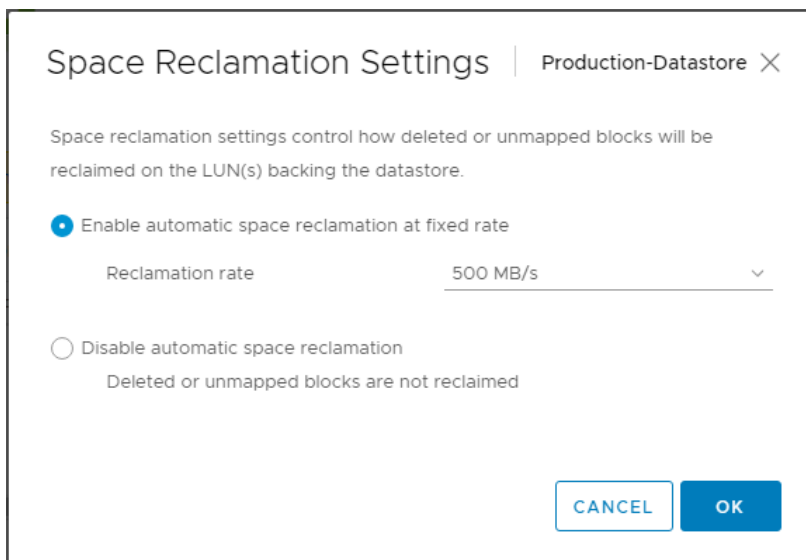


Figure 38 vSphere 6.7 client (HTML5) space reclamation settings

16.3.5 Automatic dead space reclamation

For automatic space reclamation to work, there are several requirements:

- The host must be running ESXi 6.5 and vCenter 6.5 or later.
- The datastore must be formatted with VMFS-6.
- The VMFS-6 datastore volumes must be stored within a 512k pagepool. VMware only supports automatic unmap on array block sizes less than 1 MB.
- The SC Series array must be running a SCOS version that supports VAAI UNMAP. See the [VMware HCL](#) for compatible versions.

Note: Array snapshots containing freed blocks will not release space until the snapshot expires.

16.3.6 In-guest space reclamation

For in-guest space reclamation to work, there are several requirements:

- The host must be running ESXi 6.0 and vCenter 6.0 or later. The datastore must be formatted with VMFS-6 or VMFS-5. For VMFS-5 datastores, the advanced system setting **VMFS3.EnableBlockDelete** must be set to 1.
- The datastore volumes must be stored within a 512k pagepool. VMware only supports automatic space reclamation on array block sizes less than 1 MB.
- The SC Series array must be running a SCOS version that supports VAAI UNMAP. See the [VMware HCL](#) for compatible versions.
- The virtual machine hardware must be version 11 or higher.
- The guest operating system must support UNMAP. For example: Windows 2012 R2 and higher.
- The virtual disks must be in the thin format.

Note: On datastore volumes stored with a 2 MB or greater pagepool, the volume can be manually unmapped after in-guest space reclamation using the **esxcli storage vmfs unmap --volume-label [datastore_name]**.

16.3.7 Thin provisioning stun

The thin provisioning stun primitive allows the SC Series array to send a special SCSI sense code back to the ESXi host when there is an out of space (OOS) condition. ESXi will pause only the virtual machines that are requesting additional pages until additional storage is added to remedy the situation.

For more information about VAAI primitives, see sections, “Array Thin Provisioning and VMFS Datastores” and “Storage Hardware Acceleration” in the appropriate *vSphere Storage Guide* at [VMware vSphere documentation](#).

16.4 vStorage APIs for Storage Awareness (VASA)

The vStorage APIs for Storage Awareness (VASA) are a set of protocols, routines, and tools encapsulated into an API that enable vCenter to detect the onboard capabilities of storage arrays. vCenter can obtain all of the properties of a volume such as its RAID level, performance capabilities, if it is replicated, and if Data Progression is enabled. In turn, vCenter uses these capabilities to enable profile-driven storage, allowing the administrator to determine if a virtual machine meets the organization's compliance policies.

VASA 2.0 is also required for VMware vSphere Virtual Volumes™ (vVols). vVols use the VASA provider to perform the out-of-band tasks required for the volume-to-container relationship management.

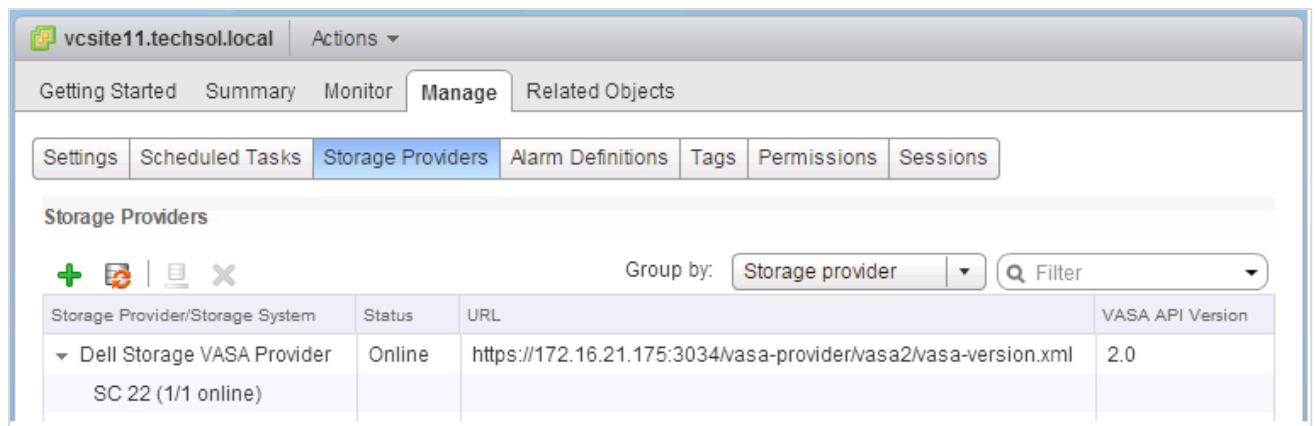


Figure 39 Adding a VASA 2.0 storage provider to vCenter

The SC Series VASA 2.0 provider is an integrated component of the DSM Data Collector, and supersedes the VASA 1.0 provider built into the CITV or DSITV appliance.

16.5 Virtual Volumes (vVols)

vVols with vSphere 6.x offer a new approach to managing storage that delivers more granular storage capabilities to the VMDK level. To accomplish this task, VMware introduced new concepts such as storage containers, protocol endpoints, and storage providers. This new vVols framework delivers software defined storage and storage policy-based management capabilities to traditional arrays.

For more detailed information about VMware Virtual Volumes, read the VMware document, [What's New: vSphere Virtual Volumes](#).

With these new foundational changes in mind, there are additional best practices that need to be followed.

- VMware vCenter Server and the DSM Data Collector servers need to be made highly available. Any disruption in services may affect availability of virtual machines stored within virtual volumes.
- When using vSphere HA, restart priority should be modified so that vCenter and DSM are brought up first.
- vCenter and DSM virtual machines must be stored on traditional datastores residing on SAN storage. Do not store or migrate these virtual machines onto a vVol datastore.
- Take regular array snapshots to protect vCenter and the VASA provider within DSM.
- Ensure proper NTP time synchronization between vCenter, ESXi, DSM, and the arrays.

Caution: When using Virtual Volumes, the VASA 2.0 provider contained within DSM becomes a critical piece of infrastructure. The DSM VM needs to be protected from outages and loss of data like any other business-critical enterprise application. Without the VASA provider active, vCenter cannot bind or unbind virtual volumes to or from the protocol endpoints. In addition, if the VASA provider is lost completely, recovery of virtual disks from within the storage container will become an arduous task.

For more information about the requirements for configuring SC Series storage and DSM for Virtual Volumes, read the following documents before beginning the implementation:

- [Dell Storage Manager 2020 R1 Administrators Guide](#)
- [Dell EMC SC Series Virtual Volumes Best Practices](#)

A Determining the appropriate queue depth for an ESXi host

Adjusting the queue depth on ESXi hosts is a complicated subject. On one hand, increasing it can remove bottlenecks and help to improve performance (if there are enough back-end disks to handle the incoming requests). However, if set improperly, the ESXi hosts could overdrive the controller front-end ports or the back-end disks and potentially make the performance worse.

The general guideline is to set the queue depth high enough to achieve an acceptable number of IOPS from the back-end disks. At the same time, it should not be set too high to allow an ESXi host to flood the front or back end of the array. The following sections provide a few basic pointers.

A.1 Fibre Channel

With 2 Gb SC Series front-end (FE) ports:

- Each 2 Gb FE port has a maximum queue depth of 256, and care must be taken to not overdrive it.
- It is best to leave the ESXi queue depths set to default and only increase them if necessary.
- The recommended settings for controllers with 2 Gb FE ports are:
 - HBA BIOS = 255 (if available)
 - Driver module = 32 (default) (The driver module regulates the HBA Queue depth)
 - DSNRO = 32 (default)
 - Guest vSCSI controller = 32 (default)

With 4/8/16 Gb SC Series front-end ports:

- Each 4/8/16 Gb front-end port can accept many more (~1900+) outstanding I/Os.
- Since each FE port can accept more outstanding I/Os, the ESXi queue depths can be set more flexibly to accommodate guest I/O. Keep in mind, the queue depth may need to be decreased if the front-end ports become saturated, the back-end disks become maxed out, or the latencies become too high.
- The recommended settings for controllers with 4/8/16 Gb FE ports are:
 - HBA BIOS = 255 (if available)
 - Driver module = 255
 - DSNRO = 32 (default) (increase or decrease as necessary)
 - Guest vSCSI controller = 32 (default) (increase or decrease as necessary)

A.2 iSCSI

With 1 Gb SC Series front-end ports, leave the queue depth set to default and only increase if necessary.

With 10 Gb SC Series front-end ports, use the following settings:

- HBA BIOS (if using hardware iSCSI) = 255
- Driver Module (iscsi_vmk) = 255
- DSNRO = 32 (default) (increase or decrease as necessary)
- Guest vSCSI controller = 32 (default) (increase or decrease as necessary)

A.3 Using esxtop to monitor queue depth

The best way to determine the appropriate queue depth is by using the esxtop utility. This utility can be ran from the ESXi Shell through SSH (command: esxtop), or vCLI 5.x or the vMA virtual appliance (command: resxtop or resxtop.sh).

When opening the esxtop utility, the best place to monitor queue depth and performance is from the Disk Device screen. Use the following steps to go to the screen:

1. From the command line, enter one of these two options:
 - # esxtop
 - resxtop.sh --server esxserver.domain.local
2. Press **u** to open the **Disk Device** screen.
3. Type **L 36** and press Enter to expand the devices field. This option expands the disk devices columns to display the volume's naa identifier.
4. Press **f** and choose the fields to monitor:
 - a. Press **b** to clear the ID field (not needed).
 - b. Optionally (depending on preference):
 - i. Check or clear **i** for overall latency.
 - ii. Check **j** for read latency.
 - iii. Check **k** for write latency.
 - c. Press [Enter] to return to the monitoring screen.
5. Type **s 2** and press [Enter] to set the refresh time to every 2 seconds.

The quick way to see if the queue depth is set correctly is to monitor the queue depth section along with the latency section, as shown in Figure 40.

DEVICE	DQLEN	ACTV	QUED	%USD	LOAD	CMD5/s	DAVG/cmd	KAVG/cmd	GAVG/cmd	QAVG/cmd
naa.600728a	32	32	0	100	1.00	1659.27	19.00	0.00	19.00	0.00
naa.600728b	32	32	16	100	1.50	1675.51	18.98	9.34	28.32	9.33
naa.600728c	32	32	32	100	2.00	1691.26	18.84	18.70	37.55	18.70
naa.600728d	32	32	96	100	4.00	1674.53	19.03	56.93	75.96	56.93

Figure 40 Esxtop with a queue depth of 32

If the LOAD is consistently greater than 1.00, and the latencies are still acceptable, the back-end disks have available IOPS so increasing the queue depth may make sense. However, if the LOAD is consistently less than 1.00, and the performance and latencies are acceptable, then there is usually no need to adjust the queue depth.

In Figure 40, the device queue depth is set to 32. Three of the four volumes consistently have a LOAD above 1.00. If the back-end disks are not maxed out, it may make sense to increase the queue depth and the DSNRO setting.

As shown in Figure 41, increasing the queue depth from the previous example has increased the total IOPs from 6700 to 7350. However, the average device latency (DAVG/cmd) increased from 18 milliseconds to 68 milliseconds. That means the latency more than tripled for a mere 9% performance gain. In this case, it may not make sense to increase the queue depth because latencies became too high. In a system that is sized with the appropriate number of drives to handle the application load, the average response time (latency) should remain below 20 milliseconds.

DEVICE	DQLEN	ACTV	QUED	%USD	LOAD	CMDS/s	DAVG/cmd	KAVG/cmd	GAVG/cmd	QAVG/cmd
naa.600728a	255	128	0	50	0.50	1847.55	67.65	0.00	67.66	0.00
naa.600728b	255	128	0	50	0.50	1852.50	67.35	0.00	67.35	0.00
naa.600728c	255	128	0	50	0.50	1837.16	68.45	0.00	68.45	0.00
naa.600728d	255	123	0	48	0.48	1813.40	67.28	0.00	67.28	0.00

Figure 41 Queue depth increased to 255

For more information about the disk statistics in esxtop, consult the esxtop man page or the *vSphere Monitoring and Performance Guide* on [VMware vSphere documentation](#).

Note: Although the per-LUN queue depth maximum is 256, the per-adapter maximum within ESXi is 4096. By increasing the per LUN queue depth from 64 to 128, it can take fewer LUNs to saturate a port queue. For example, $4096/64=64$ LUNs, but $4096/128=32$ LUNs.

B Deploying vSphere client plug-ins

With SC Series storage, multiple vSphere client plug-ins are available to aid in administration and management of the arrays. Depending on the version, these plug-ins allow administrators to provision datastores, take snapshots, create replications, and even report usage and performance statistics directly from within the vSphere client.

B.1 Dell Storage vSphere Web Client plug-in

The web client plug-in is contained within the Dell Storage Integration for VMware DSITV (formerly CITY) virtual appliance. Once the appliance is deployed and given an IP address, the web plug-in can be registered with vCenter, and administrators can provide their DSM login credentials to manage all their arrays.

The DSITV virtual appliance can be downloaded from the [Dell Support](#) site.

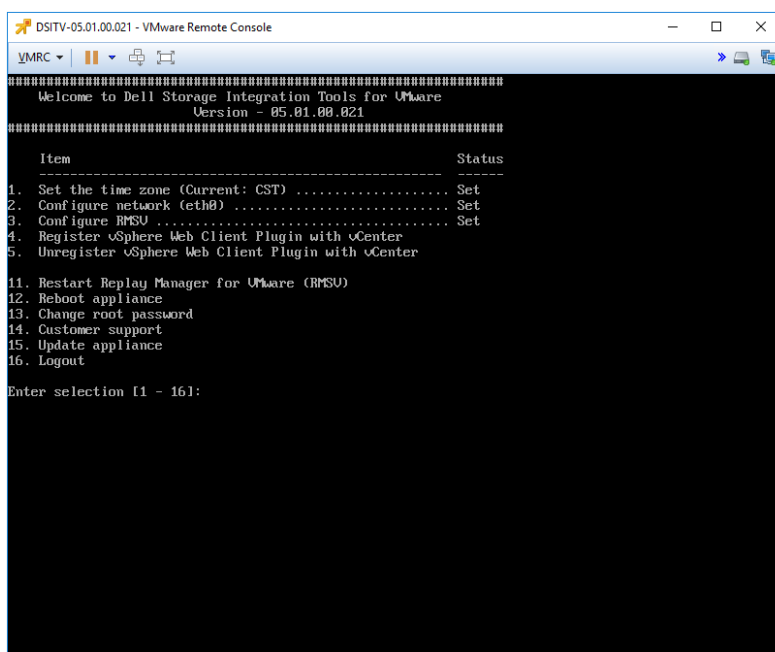


Figure 42 DSITV 5.x virtual appliance configuration screen

C Configuring Dell Storage Manager VMware integrations

With DSM, the Data Collector can be configured to gather storage statistics and perform basic storage administration functions with vCenter. This functionality is also used to configure and enable vVols.

To add vCenter credentials into the Data Collector, enter the **Servers** viewer screen, right-click **Servers**, and select **Register Server**.

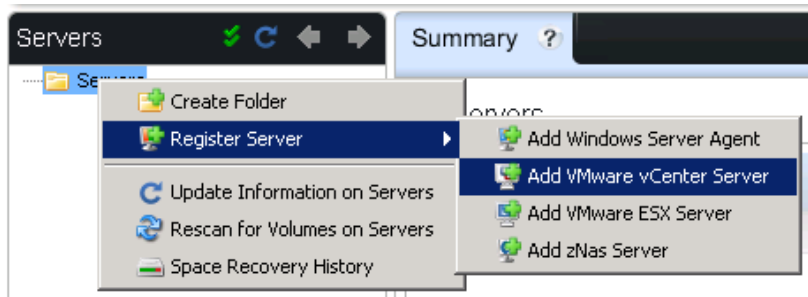


Figure 43 Adding a vCenter Server

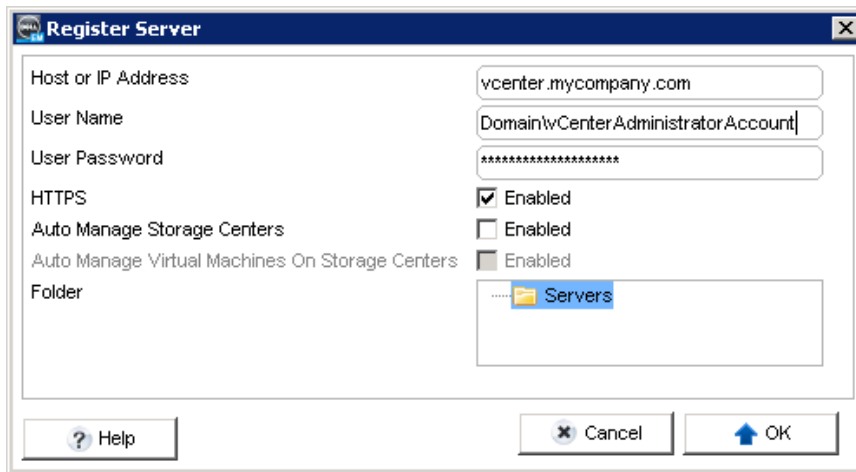


Figure 44 Registering vCenter server credentials

After entering vCenter credentials, administrators can see aggregate storage statistics and provision VMFS datastores and RDMS. For example, when creating a volume, selecting an ESXi host automatically gives the option to format it with VMFS. Similarly, when creating a volume to be assigned to a virtual machine, DSM can automatically add the volume as a pRDM.

D Host and cluster settings

This section summarizes the host settings for each of the storage protocols. At the time of this writing, these settings were valid for vSphere 7.x running on SCOS 6.6 and later. For each setting, see the referenced section number to determine if the setting is applicable for your environment.

D.1 Recommended settings

Apply these settings to each of the ESXi hosts:

- **Fibre Channel: QLogic driver module settings for queue depth and controller failover (section 4.2.1)**

```
esxcli system module parameters set -m qlnativefc -p "ql2xmaxqdepth=255
ql2xloginretrycount=60 qlport_down_retry=60"
```

- **Fibre Channel: Emulex driver module settings for queue depth and controller failover (section 4.2.1)**

```
esxcli system module parameters set -m lpfc820 -p "lpfc_devloss_tmo=60
lpfc_lun_queue_depth=254"
```

- **All protocols: Setting round robin as the default path selection policy (section 6.9.1.1)**

```
esxcli storage nmp satp rule add -s VMW_SATP_ALUA -V COMPELNT -P
VMW_PSP_RR -o disable_action_OnRetryErrors -e "Dell EMC SC Series Claim
Rule"
```

- **SAS: Configuring the recommended module parameters for the SAS driver (section 4.2.3)**

```
esxcli system module parameters set -p issue_scsi_cmd_to_bringup_drive=0
-m lsi_msgpt3
```

- **FCoE: Setting the ALUA module default for FCoE volumes**

```
esxcli storage nmp satp rule add -R fcoe -s VMW_SATP_ALUA
```

- **iSCSI: Setting ARP redirect for hardware iSCSI adapters (section 3)**

```
esxcli iscsi physicalnetworkportal param set --option ArpRedirect -v=1 -A
vmhbaXX
```

- **iSCSI: Software iSCSI Queue Depth (section 4.2.2)**

```
esxcli system module parameters set -m iscsi_vmk -p iscsivmk_LunQDepth=255
```

- **iSCSI: Software iSCSI login timeout (section 4.2.2)**

```
esxcli iscsi adapter param set -A=vmhbaXX -k=LoginTimeout -v=5
```

- **iSCSI: Disable Delayed ACK (section 3)**

```
esxcli iscsi adapter param set -A vmhbaXX -k DelayedAck -v false
```

- **All protocols: HA Cluster Settings (section 4.3)**

```
esxcli system settings kernel set -s terminateVMonPDL -v TRUE
```

- All protocols: HA Cluster Settings (section 4.3)

```
esxcli system settings advanced set -o "/Disk/AutoremoveOnPDL" -i 1
```

- All protocols: Advanced options HA Cluster setting (section 4.3)

```
das.maskCleanShutdownEnabled = True
```

D.2 Optional settings

These settings are optional but are listed because they solve certain corner-case performance issues. Settings in this section should be tested to ascertain if the settings improve function.

- All protocols: DSNRO per datastore (section 4.4)

```
esxcli storage core device set -d <naa.dev> -O <value of 1-256>
```

- All protocols: Conditions for round robin path changes, individual datastore (section 6.9.1.2)

```
esxcli storage nmp psp roundrobin deviceconfig set --device naa.XXX --  
type=iops --iops=3
```

- All protocols: Conditions for round robin path changes, all datastores on host (section 6.9.1.1)

```
esxcli storage nmp satp rule add -s VMW_SATP_ALUA -V COMPELNT -P  
VMW_PSP_RR -o disable_action_OnRetryErrors -e "Dell EMC SC Series Claim  
Rule" -O "policy=iops;iops=3"
```

E Additional resources

E.1 Technical support and resources

[Dell.com/support](https://www.dell.com/support) is focused on meeting customer needs with proven services and support.

[Storage technical documents and videos](#) provide expertise that helps to ensure customer success on Dell EMC storage platforms.

E.2 VMware support

For VMware support, see the following resources:

- [VMware vSphere Documentation](#)
- [VMware Knowledge Base](#)
- [VMware.com](https://www.vmware.com)
- [Education and training](#)
- [VMware communities](#)