Getting To Know Dell EMC VxFlex OS

Version 3.x

Rev 03

September 2019

DELL EMC

Copyright © 2019 Dell Inc. or its subsidiaries. All rights reserved.

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

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS-IS." DELL 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 DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

Dell Technologies, Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be the property of their respective owners. Published in the USA.

Dell EMC
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.DellEMC.com

CONTENTS

Figures 5

Tables 7

Preface 9

Part 1 Getting to Know VxFlex OS 11

Chapter 1 What is VxFlex OS? 13
Overview of VxFlex OS....14
Overview of VxFlex Ready Node....14
Workflow of VxFlex OS....14

Part 2 Architecture 17

Chapter 2 Architecture 19
System....20
Hardware....20
Software....20
VxFlex OS implementation in an ESXi-based system....21
Prerequisites for adding an NVDIMM to an ESXi 6.7 system....23
Verify that the ESXi host can detect the NVDIMM....23
Calculate required NVDIMM and RAM capacity for FG SDS....24
The MDM cluster....24
Storage definitions....27
Protection Domains....27
Fault Sets....27
Storage Pools....28
Acceleration Pools....30
Data layout....30
Volumes....31
Naming....31
Protection and load balancing....31
Rebuild....31
Rebuild throttling....32
Rebalance....33
Rebalance throttling....33
Checksum protection....34
ESXi vStorage APIs for Array Integration (VAAI)....34
Cache....35
Networking....38
VMware limitation....41
Virtual IP Address....41
Monitor of SDC and SDS connections....42
S.M.A.R.T. hardware monitoring....43
List of approved RAID controllers....43

Snapshots....44
V-Trees....45
Snapshot Policies....46
Performance....47
Migrate volumes....47
Other functions....47
Implementing VxFlex OS....51
Implementing VxFlex OS over a virtual system....54
VxFlex OS implementation in an ESXi-based system....54
Xen implementation....56
Maintenance....57
Maintain the physical layer....57
Instant maintenance mode....57
Maintaining the virtualization layer....58
Management tools....58
VxFlex OS VASA architecture....59
VxFlex OS VASA's limitations and prerequisites....59
Configuring direct attached storage (DAS)....60
Product capabilities and system limits....61

Part 3 System Requirements 63

Chapter 3 System Requirements 65
System requirements....66
VxFlex OS cluster components....66
Physical server requirements....66
Supported operating systems....68
SVM memory requirements....69
External SDC support....70
VxFlex OS GUI server requirements....70
VxFlex OS Gateway server requirements....71
Other requirements....71
License for VxFlex OS....72
Activate the license....72
Activating entitlements and installing a license file....74
Activate an entitlement and download the license file....74
Install the license....77
License file example....78
Licensing error messages....78

FIGURES

1 VxFlex OS implementation on ESXi....21
2 5-node MDM cluster....26
3 Protection Domains, Storage Pools, and Fault Sets....28
4 Protection Domains and Storage Pools....29
5 VxFlex OS deployed on a single network....40
6 VxFlex OS deployed on separate networks....41
7 Snapshot operations....45
8 V-Tree diagram....46
9 Physical layout example—3-node cluster....52
10 VxFlex OS implementation on ESXi....54
11 VxFlex OS Xen virtual machine architecture....56
12 Licensing LAC email....74
13 License file example....78

TABLES

1 MDM cluster modes....26
2 Caching modes....36
3 Caching support matrix....38
4 IP address configurations in VxFlex OS (based on CSV file)....39
5 Product limits....61
6 Server physical requirements....66
7 Supported operating systems -VxFlex OS components....68
8 eLicensing terminology(continued)....73
9 Licensing error messages....78

Preface

As part of an effort to improve its product lines, Dell EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your Dell EMC technical support professional if a product does not function properly or does not function as described in this document.

Note: This document was accurate at publication time. Go to Dell EMC Online Support (https://support.emc.com) to ensure that you are using the latest version of this document.

Previous versions of Dell EMC VxFlex OS were marketed under the name Dell EMC ScaleIO.

Similarly, previous versions of Dell EMC VxFlex Ready Node were marketed under the name Dell EMC ScaleIO Ready Node.

References to the old names in the product, documentation, or software, etc. will change over time.

Note: Software and technical aspects apply equally, regardless of the branding of the product.

Related documentation

The release notes for your version includes the latest information for your product.

The following Dell EMC publication sets provide information about your VxFlex OS or VxFlex Ready Node product:

- VxFlex OS software (downloadable as VxFlex OS Software <version> Documentation set)
- VxFlex Ready Node with AMS (downloadable as VxFlex Ready Node with AMS Documentation set)
- VxFlex Ready Node no AMS (downloadable as VxFlex Ready Node no AMS Documentation set)
- VxRack Node 100 Series (downloadable as VxRack Node 100 Series Documentation set)

You can download the release notes, the document sets, and other related documentation from Dell EMC Online Support.

Typographical conventions

Dell EMC uses the following type style conventions in this document:

Bold	Used for names of interface elements, such as names of windows, dialog boxes, buttons, fields, tab names, key names, and menu paths (what the user specifically selects or clicks)
Italic	Used for full titles of publications referenced in text
`Monospace`	Used for: • System code • System output, such as an error message or script • Pathnames, filenames, prompts, and syntax • Commands and options
Monospace italic	Used for variables

`Monospace bold`	Used for user input
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections - the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

Dell EMC support, product, and licensing information can be obtained as follows:

Product information

For documentation, release notes, software updates, or information about Dell EMC products, go to Dell EMC Online Support at https://support.emc.com.

Technical support

Go to Dell EMC Online Support and click Service Center. You will see several options for contacting Dell EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your Dell EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to techpubcomments@emc.com.

PART 1

Getting to Know VxFlex OS

The following section provides an overview of VxFlex OS.

Chapter 1, "What is VxFlex OS?"

CHAPTER 1

What is VxFlex OS?

What is VxFlex OS and how does it work?

VxFlex OS is a software-only solution that uses existing servers' local drives and LAN to create a virtual SAN that has all the benefits of external storage. VxFlex OS utilizes the existing local storage devices and turns them into shared block storage.

VxFlex OS software components are installed on the application servers and communicate via a standard LAN to handle the application I/O requests sent to VxFlex OS block volumes. An extremely efficient decentralized block I/O flow, combined with a distributed, sliced volume layout, results in a massively parallel I/O system that can scale up to thousands of nodes.

- Overview of VxFlex OS 14
- Workflow of VxFlex OS 14

Overview of VxFlex OS

VxFlex OS is a software-only solution that uses existing servers' local disks and LAN to create a virtual SAN that has all the benefits of external storage—but at a fraction of cost and complexity. VxFlex OS utilizes the existing local storage devices and turns them into shared block storage. For many workloads, VxFlex OS storage is comparable to, or better than external shared block storage.

The lightweight VxFlex OS software components are installed on the application servers and communicate via a standard LAN to handle the application I/O requests sent to VxFlex OS block volumes. An extremely efficient decentralized block I/O flow, combined with a distributed, sliced volume layout, results in a massively parallel I/O system that can scale up to thousands of nodes.

VxFlex OS is designed and implemented with enterprise-grade resilience. Furthermore, the software features an efficient distributed self-healing process that overcomes media and server failures, without requiring administrator involvement.

Dynamic and elastic, VxFlex OS enables administrators to add or remove servers and capacity on-the-fly. The software immediately responds to the changes, rebalancing the storage distribution and achieving a layout that optimally suits the new configuration.

Because VxFlex OS is hardware agnostic, the software works efficiently with various types of disks, including: magnetic (HDD) and solid-state disks (SSD), flash PCI Express (PCIe) cards, networks, and hosts.

VxFlex OS can easily be installed in an existing infrastructure as well as in green field configurations.

Overview of VxFlex Ready Node

VxFlex Ready Node is the combination of VxFlex OS software-defined block storage and Dell PowerEdge® servers, optimized to run VxFlex OS, enabling customers to quickly deploy a fully architectured, software-defined, scale out server SAN.

Any hypervisor can run on VxFlex Ready Node servers as an application consuming VxFlex OS volumes.

In an AMS-based solution, a limited number of ESXi and RHEL operating systems are currently supported. For more information, see the operating system support tables.

The solution is managed by the AMS (Automated Management Services), which enables a simple or customized deployment process from bare metal, no IP state, to a fully-configured system: IP address assignment, VxFlex OS deployment and configuration, and vCenter configuration.

Note: In the documentation set, there are references that are specific to either ESXi or RHEL operating systems, but not to both (for example, vCenter). These differences are not marked in most places.

Workflow of VxFlex OS

The following steps describe the order to follow to get the VxFlex OS system up and running in your environment:

1. Assemble your hardware according to your requirements for your environment.
2. Connect and configure the network.
3. Install the Operating System and prerequisites for VxFlex OS.
4. Deploy VxFlex OS. Refer to the *Deployment Guide* for step by step deployment procedures.
5. Allocate volumes and map them to hosts.

6. Begin using the VxFlex OS storage.

Here is a list of other VxFlex OS functions that you may require while setting up your system:

- Implementing VxFlex OS .
- Implementing VxFlex OS over a virtual system.
- Implementing VxFlex OS in an ESXi-based system.
- Implementing VxFlex OS on a XEN system.

PART 2

Architecture

This section describes the architecture of the VxFlex OS system.

Chapter 2, "Architecture"

CHAPTER 2

Architecture

The following topics describe the different components of the VxFlex OS system.

- System 20
- VxFlex OS implementation in an ESXi-based system 21
- The MDM cluster 24
- Storage definitions 27
- Protection and load balancing 31
- ESXi vStorage APIs for Array Integration (VAAI) 34
- Cache 35
- Networking 38
- Virtual IP Address 41
- Monitor of SDC and SDS connections 42
- S.M.A.R.T. hardware monitoring 43
- Snapshots 44
- Performance 47
- Migrate volumes 47
- Other functions 47
- Implementing VxFlex OS over a virtual system 54
- Maintenance 57
- Management tools 58
- VxFlex OS VASA architecture 59
- Configuring direct attached storage (DAS) 60
- Product capabilities and system limits 61

System

VxFlex OS is based on a hardware and a software component.

Hardware

This section outlines the VxFlex OS hardware components.

In general, hardware can be the existing application servers used by the datacenter, or a new set of nodes (if, for example, you want to dedicate all nodes solely for the purpose of running the VxFlex OS SAN storage system).

The VxFlex Ready Node system includes a full datacenter solution including: rack, switches, server nodes , power and cabling. Built for capacity, performance, or compute per the specific configuration.

- *Nodes*
  Nodes, or servers, are the basic computer unit used to install and run VxFlex OS. They can be the same servers used for the applications (server convergence), or a dedicated cluster. In any case, VxFlex OS is hardware-agnostic, and therefore, aside from performance considerations, the type of server is inconsequential.
- *Storage Media*
  The storage media can be any storage media, in terms of the type (HDD, SSD, NVMe SSD or PCIe flash cards) and anywhere (DAS, or external).

Software

The VxFlex OS virtual SAN software consists of a Meta Data Manager (MDM), a Storage Data Server (SDS) and a Storage Data Client (SDC).

The VxFlex OS virtual SAN consists of the following software components:

- Meta Data Manager (MDM)
  Configures and monitors VxFlex OS. The MDM can be configured in redundant cluster mode, with three members on three servers or five members on five servers, or in single mode on a single server.

  NOTICE It is not recommended to use single mode in production systems, except in temporary situations. The MDMs contains all the metadata required for system operation. single mode has no protection, and exposes the system to a single point of failure.
- Storage Data Server (SDS)
  Manages the capacity of a single server and acts as a back-end for data access. The SDS is installed on all servers contributing storage devices to VxFlex OS. These devices are accessed through the SDS.
- Storage Data Client (SDC)
  A lightweight device driver that exposes VxFlex OS volumes as block devices to the application that resides on the same server on which the SDC is installed.

Depending on the desired configuration (described later), the software components are installed on the server and give rise to a virtual SAN layer exposed to the applications that reside on the servers.

VxFlex OS implementation in an ESXi-based system

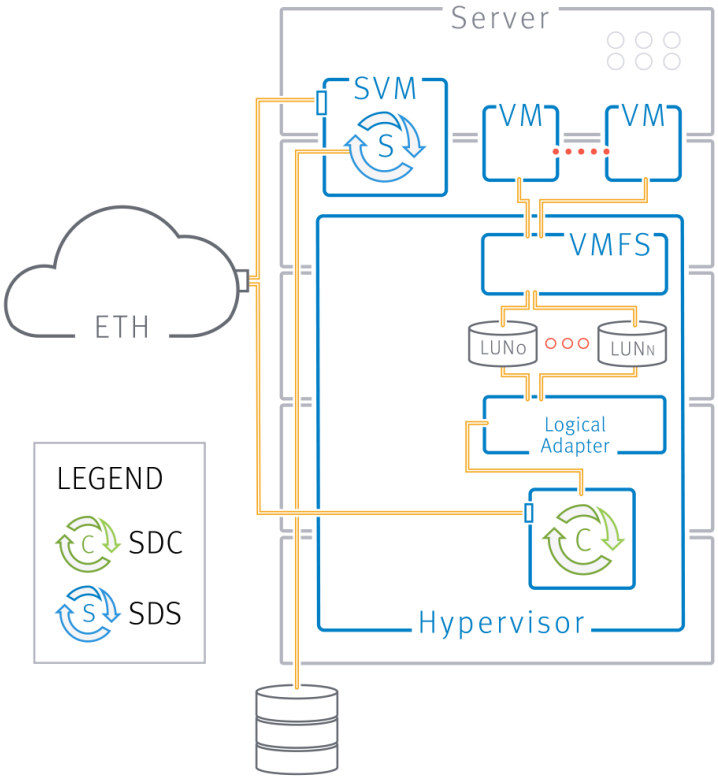
In the VMware environment, the MDM and SDS components are installed on a dedicated SVM, whereas the SDC is installed directly on the ESXi host.

Implementation

Note:
Installing the SDC on the ESXi host requires a restart of the ESXi host.

This implementation is illustrated in the following figure:

Figure 1 VxFlex OS implementation on ESXi



The LUNs in the previous figure can be formatted with VMFS, and then exposed using the ESXi host to the virtual machine, or can be used as RDM devices. When the LUNs are used as RDM devices, the VMFS layer is omitted.

Installation in a VMware environment is performed using the vSphere plug-in.

Device management

Devices can be managed in the following ways:

- VMDirectPath I/O
  Device management is performed via the SVM, yielding the best I/O performance. Devices are added to the system after the deployment. On factory-installed VxFlex Ready Node servers, the vSphere plug-in configures DirectPath on the ESXi servers and adds devices to the system. On other servers, you must enable VMDirectPath manually to the ESXi server, then use the plug-in to add devices.

Requirements:

- When DirectPath is configured, all servers in the system must use DirectPath. For VxFlex Ready Node or VxRack Node 100 Series systems, this can be configured with the VxFlex OS vSphere plug-in. For other servers, use the vSphere client to configure each ESXi host manually.
- If the host has multiple controllers, you must configure DirectPath on that host manually, not with the plug-in. After that is done, you can use the plug-in to deploy VxFlex OS on that host and to add devices to it.
- All devices on all servers must not have any other use (not VMDK, RDM, or be part of a datastore).
- PowerEdge 14G servers with any NVME devices cannot be added in a DirectPath-based system. Use RDM-based, instead.
- ESXi boot device requirements:
  - Must be on a separate controller or connected directly to the board.
  - SATADOM and M2 boot devices are supported. These devices allow the creation of a datastore on the system disk, which is needed to host the SVM.
  - USB boot devices are not supported.

Note: DirectPath architecture is the recommended best practice for RAID and SAS Controller managed drives.

- RDM

  Using RDM mapping, a device is created on the SVM that points to the physical disk on the ESXi.

  You can add RDM devices that are connected through a physical RAID controller. If a local RDM is not connected via a RAID controller, it may not be supported. To ensure the compatibility of these devices, you can add them as VMDK, or you can select **Force RDM on device with unknown RDM support**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*. Enable this option before beginning the deployment.

  Before enabling this feature, contact Customer Support.

- VMDK

  A new datastore is created, with a VMDK, and the VMDK is added to the SVM. VxFlex OS requires thick provisioning, so this process can take a long time.

  In almost all cases, RDM is the preferred method to add physical devices. Use the VMDK method only in the following scenarios:

  - The physical device does not support RDM.
  - The device already has a datastore, and the device isn't being completely used. The excess area that is not already being used will be added as a VxFlex OS device.

    Note: To use VMDK, select **Enable VMDK creation**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*.

System size

If you are deploying a very large VxFlex OS system (hundreds of nodes), you can increase the default parallelism limit (default: 100), thus speeding up the deployment. This is dependent on the processing power of the vCenter.

To increase the parallelism limit, use the plug-in **Advanced settings**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*.

Pre-deployment considerations

You should take these considerations into account before deploying the system:

- Do you want to use separate networks for data and management (recommended), and which IP addresses will you use for the SVMs and VMkernels?
- Are there flash devices that will be added to the SDS?
- Do you want to create Fault Sets? See the requirements in "Fault Sets".

Post-deployment considerations

You should take these considerations into account after deploying the system:

- After deployment is complete, set all SVMs to start automatically with the system. Do not set SVMs under the VMware resource-pool feature.
- In a DirectPath environment, after deploying the system, you must add devices to the SDS.

Other vSphere plug-in features

The plug-in's deployment wizard can be used to register existing VxFlex OS systems in vSphere, to add SDS server nodes, or to extend the existing system.

The plug-in can be used to provision VxFlex OS nodes.

Note: When using DirectPath, also known as PCI-passthrough on the SVM, the following alert appears in vCenter: "Virtual Machine memory Usage". This is a VMWare limitation and can't be resolved. Refer to the VMWare KB for more details:https://kb.vmware.com/s/article/2149787

Prerequisites for adding an NVDIMM to an ESXi 6.7 system

Meet the following prerequisites before adding an NVDIMM to an ESXi 6.7 based VxFlex OS system.

Before you begin adding an NVDIMM to an ESXi 6.7 based VxFlex OS system, ensure the following:

- You are using VMware version 6.7 for the ESXi host and the vCenter server.
- The VM version of your SVM is version 14 or higher.
- The firmware of the NVDIMM is updated to the version supported by your server. For example, for compatible Dell EMC PowerEdge servers, use NVDIMM firmware version 9324 or higher.
- The ESXi host can detect the NVDIMM.
- You calculate required NVDIMM capacity, according to number of FG devices and their size.
- There is enough NVDIMM capacity on the ESXi host.

Verify that the ESXi host can detect the NVDIMM

Before adding an NVDIMM to an ESXi-based VxFlex OS system, verify that the ESXi host can detect the NVDIMM.

Before you begin

Procedure

1. Login to the vCenter.
2. Select your ESXi host.
3. Navigate to the **Summary** tab.
4. In the **Hardware** section, verify that the required amount of **Persistent Memory** is listed.

Calculate required NVDIMM and RAM capacity for FG SDS

Use the following formulas to calculate the necessary capacity for NVDIMM and RAM.

About this task

Note: When adding NVDIMM capacity to the SVM, the complete capacity of the NVDIMM must be allocated in a single device when adding to the SVM. For example 16 GB*2 of NVDIMM will be added as 31 GB (rounded down, since VMware uses some of the capacity for datastore management).

Procedure

1. Calculate the required NVDIMM and RAM capacity using the following table, where *X* is the total capacity of SDS devices used for the FG storage pool, and *Y* is the total RAM needed:

FG capacity	Required NVDIMM capacity	Required RAM capacity (rounded)	Additional memory for MDM, LIA, and SVM OS	Total RAM (rounded)
51.2 TB	32 GB NVDIMM (in SVM it will be 31 GB)	>41 GB	MDM: 5.4 GB LIA: 350 MB OS Base: 1 G Buffer : 1 G	53 GB
51.2 TB < *X* < 96 TB	64 GB NVDIMM (in SVM it will be 62 GB)	>41 GB <64.5 GB		53 GB < *Y* < 73 GB
96 TB < *X* < 128 TB (122.88 is actual limit)	96 GB NVDIMM (in SVM it will be 93 GB)	>64.5 GB <81.5 GB		87 GB

2. Alternatively, you can calculate NVDIMM capacity and RAM capacity using the following formulas:

   Note:
   The calculation is in binary MiB, GiB, and TiB

   - *NVDIMM_capacity_in_GiB* = ((100 * *Number_Of_Drives*) + (700 * Capacity_in_TiB))/1024
   - *RAM_capacity_in_GiB* = 10 + ((100 * *Number_Of_Drives*) + (550 * Capacity_in_TiB))/1024

The MDM cluster

The MDM functions as the monitoring and configuration agent of VxFlex OS and is used mainly for management. A multi-MDM environment consists of one Master MDM, while the others function as Slaves or Tie-Breakers.

The MDM serves as the monitoring and configuration agent of VxFlex OS. The MDM is mainly used for management which consists of migration, rebuilds, and all system-related functions. No I/O run through the MDM.

To support high availability, three or five instances of MDM run on different servers. In a multi-MDM environment, one MDM is given the Master role, and the others act as Slave or Tie Breaker MDMs.

The MDM cluster comprises a combination of Master MDM, Slave MDMs, and Tie Breaker MDMs. There is also the Standby MDM which is not a part of the cluster.

The following terms are relevant to the MDM, the building blocks of the MDM cluster:

- MDM

  Any server with the MDM package installed. An MDM can be given a Manager or a Tie Breaker (default) role, during installation. This role can't be changed later without reinstalling the MDM. MDMs have a unique MDM ID, and can be given unique names.

  Before the MDM can be part of the cluster, it must be promoted to a Standby MDM.

- Standby MDM and Tie Breaker

  An MDM and a Tie Breaker can be added to a system as a standby. Once added, the standby MDM or Tie Breaker is attached, or locked, to that specific system.

  A standby MDM can be called on to assume the position of a Manager MDM or Tie Breaker MDM according to how it is installed, when it is promoted to be a cluster member.

- Manager MDM

  An MDM that can act as a Master or a Slave in the cluster. Manager MDMs have a unique system ID, and can be given unique names. A manager can be a standby or a member of the cluster.

  In VxFlex OS documentation, “MDM” refers to a manager, unless specified otherwise.

- Tie Breaker MDM

  An MDM whose sole role is to help determine which MDM is the master. A Tie Breaker can be a standby or a member of the cluster. A Tie Breaker MDM is not a manager.

  In a 3-node cluster, there is one TB; in a 5-node cluster, there are two TBs. This ensures that there are always an odd number of MDMs in a cluster, which guarantees that there is always a majority in electing the master.

The following terms are relevant to the MDM cluster, specifically:

- Master MDM (previously called Primary MDM)

  The MDM in the cluster that controls the SDSs and SDCs. The Master MDM contains and updates the MDM repository, the database that stores the SDS configuration, and how data is distributed between the SDSs in the system. This repository is constantly replicated to the Slave MDMs, so they can take over with no delay.

  Every MDM cluster has one Master MDM.

- Slave MDM (used to be called Secondary MDM)

  An MDM in the cluster that is ready to take over the Master MDM role if ever necessary.

  In a 3-node cluster, there is one Slave MDM, thus allowing for a single point of failure. In a 5-node cluster, there are two Slave MDMs, thus allowing for two points of failure. This increased resiliency is a major benefit to enabling the 5-node cluster.

- Replica

  An MDM that contains a replica of the MDM repository. This includes the Master MDM and any Slave MDMs in the MDM cluster.

The following table describes the available cluster modes:

Table 1 MDM cluster modes

Cluster mode	Members	Description
3-node (default)	• Master MDM • Slave MDM • Tie Breaker	3-node cluster has two copies of the repository, thus can withstand one MDM cluster failure.
5-node	• Master MDM • Two Slave MDM • Two Tie Breaker	5-node cluster has three copies of the repository, thus can withstand two MDM cluster failure.
Single-node	• Master MDM	Single-node cluster has only one copy of the repository, thus it cannot withstand failure. It is not recommended to use Single Mode in production systems.

In addition to the cluster members, you can prepare standby Managers and Tie Breaker nodes, for a total of thirteen cluster and standby MDMs.

The MDM cluster IP address limit is 16 IP addresses, which includes all cluster members (Master, Slave, Standby Master, and Standby Slaves).

The following figure illustrates a 5-node MDM cluster:

Figure 2 5-node MDM cluster

All members of the MDM cluster have the same MDM package installed on them.

Before a server makes its way into the MDM cluster, it must follow the following path:

1. Install the MDM package on the server.
   During the installation, you determine if the server will be a Manager or a Tie Breaker (default).
2. Promote the server to Standby status, either as a Manager or as a Tie Breaker according to how it was installed.
3. Add the standby server to the MDM cluster. A Manager, once entered into the cluster can take on the Master or Slave state.

MDM cluster creation is done automatically when deploying a system with any of the automated deployment tools.

Storage definitions

When configuring VxFlex OS, you should take the following concepts into account: Protection Domains, Storage Pools, and Fault Sets. Together, these elements link the physical layer to the virtual storage layer.

Protection Domains

A Protection Domain is a logical entity that consists of a group of SDSs that backs up each other. Protection Domains can be added during installation and modified post-installation.

A Protection Domain is a logical entity that contains a group of SDSs that provide backup for each other. Each SDS belongs to one (and only one) Protection Domain. Thus, by definition, each Protection Domain is a unique set of SDSs. In Figure 3 there are three Protection Domains. The one in the middle (fully depicted) consists of seven SDSs, each with two storage devices.

The maximum recommended number of nodes in a Protection Domain is 128. This enables the following:

- optimal performance
- reduction of theoretical *mean time between failure* issues
- ability to sustain multiple failures in different Protection Domains

You can add Protection Domains during installation. In addition, you can modify Protection Domains post-installation with all the management clients (except for OpenStack) .

Fault Sets

A Fault Set defines a group of SDSs that are more inclined to go offline simultaneously. Defining a Fault Set configures VxFlex OS to mirror the devices in the set.

A Fault Set is a logical entity that contains a group of SDSs within a Protection Domain, that have a higher chance of going offline together, such as a group of SDSs that are all powered in the same rack. By grouping them into a Fault Set, you are configuring VxFlex OS to mirror the data for all devices in this Fault Set. The mirroring should take place on SDSs that are outside of this Fault Set.

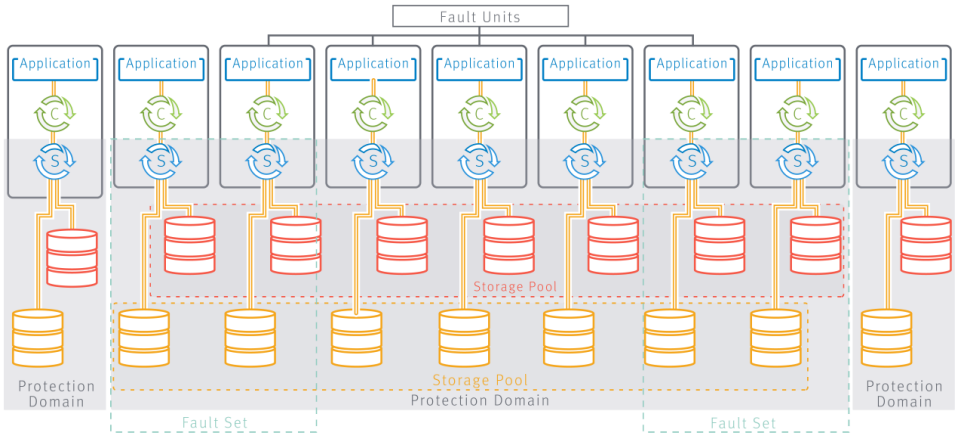
When defining Fault Sets, we refer to the term fault units, where a fault unit can be either a Fault Set, or an SDS not associated with a Fault Set (you may think of it as a Fault Set of a single SDS).

There must be enough capacity within at least three fault units to enable mirroring.

If Fault Sets are defined, you can use any combination of fault units, for example:

- SDS1, SDS2, SDS3
- FS1, SDS1, SDS2
- FS1, FS2, SDS1
- FS1, FS2, FS3

Figure 3 Protection Domains, Storage Pools, and Fault Sets



To use Fault Sets, you must work in the following order:

1. Ensure that a Protection Domain exists, or add a new one.
2. Ensure that a Storage Pool and Fault Sets (minimum of three fault units) exist, or add new ones.
3. Add the SDS, designating the Protection Domain and Fault Set, and at the same time, adding the SDS devices into a Storage Pool.
   The automated deployment and installation tools follow this order automatically.

   You can only create and configure Fault Sets before adding SDSs to the system, and configuring them incorrectly may prevent the creation of volumes. An SDS can only be added to a Fault Set during the creation of the SDS.

   You define Fault Sets and add SDSs to them during installation, using the following management tools:

- VxFlex OS Installer
- CLI
- REST
- vSphere plug-in

You can also add Fault Sets when adding SDS nodes after initial installation.

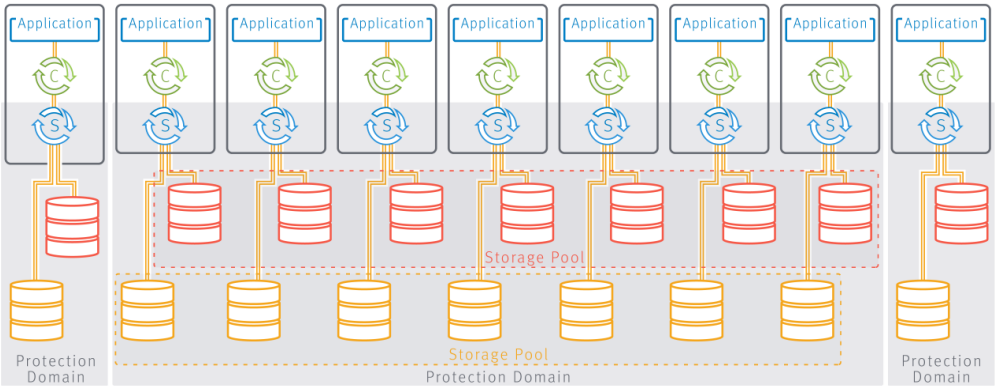
Storage Pools

A Storage Pool is a set of physical storage devices in a Protection Domain. A volume is distributed over all devices residing in the same Storage Pool. You can define a Magnetic Storage Pool (for HDDs) or a High Performance Storage Pool (for SSDs). Storage Pools support medium or fine granularity data layouts and allow enabling or disabling zero padding.

Storage Pools allow the managing of different storage tiers in VxFlex OS. A Storage Pool is a set of physical storage devices in a Protection Domain. Each storage device belongs to one (and only one) Storage Pool. The figure "Protection Domains and Storage Pools" depicts two Storage Pools.

When a volume is configured over the virtualization layer, (see "SAN virtualization layer"), it is distributed over all devices residing in the same Storage Pool. Each volume block has two copies located on two different Fault Units. This allows the system to maintain data availability following a single-point failure. Two network failures render all the domain in a state where IO errors are possible on any storage pool

Figure 4 Protection Domains and Storage Pools



You must assign a media type setting to each Storage Pool. Supported types are: HDD, SSD and Transitional (allows for migration flows).

If all SDSs in a Protection Domain have two physical drives associated with them—one HDD, and the other SSD— you should define two Storage Pools:

- Magnetic Storage Pool
  Consists of all HDDs in the Protection Domain
- High performance Storage Pool

  Consists of all SSDs used for storage purposes in the Protection Domain

  Note: Mixing different types of SSD's is not recommended and creating a separate Storage Pool for each type is the recommended best practice. For example: SAS SSD, SATA SSD, NVMe SAS SSD.

  Note:

  - VxFlex OS might not perform optimally if there are large differences between the sizes of the fault units in the same Storage Pool. For example, if one device has a much larger capacity than the rest of the devices, performance may be affected. After adding devices, you can define how much of the device capacity is available to VxFlex OS by using the SCLI `modify_sds_device_capacity` command.

Storage Pools support the following data layouts for HDD or SSD media:

- Medium Granularity (MG): space allocation occurs at 1MB units
  - suppports HDD and SSD media
- Fine Granularity (FG):
  - requires SSD media and NVDIMM for acceleration
  - space allocation occurs at 4KB units
  - includes persistent checksum for data integrity
  - supports data compression which reduces the size of data stored on the disk
  - supports thin-provisioned, zero-padded volumes

Note: FG and MG storage pools can both exist in a single SDS. You can also migrate volumes across the two layouts.

Each Storage Pool can work in one of the following modes:

- Zero-padding enabled

Ensures that every read from an area previously not written to returns zeros. Some applications might depend on this behavior. Furthermore, zero padding ensures that reading from a volume will not return information that was previously deleted from the volume.

This behavior incurs some performance overhead on the first write to every area of the volume since the area needs to be filled with zeros first.

FG is always zero padded.

- Zero-padding disabled (default only for MG)

  A read from an area previously not written to will return unknown content. This content might change on subsequent reads.

Zero padding must be enabled if you plan to use any other application that assumes that when reading from areas not written to before, the storage will return zeros or consistent data.

You can add Storage Pools during installation. In addition, you can modify Storage Pools post-installation with most of the management clients.

Note: The zero padding policy cannot be changed after the addition of the first device to a specific Storage Pool.

Acceleration Pools

Acceleration Pools enable RFCache of SSD or NVMe devices to accelerate an HDD storage pool. It also enables NVRAM required for Fine Granularity (FG) storage pools.

RFCache Acceleration Pools features and requirements:

- RFCache feature consumes the entire device, which prevents other features being used
- You can only have a single RFCache Acceleration Pool per Protection Domain
- NDU creates a default RFCache Acceleration Pool
- You can disable/enable RFCache via PD, SP, and SDS
- A storage device benefits from RFCache acceleration if RFCache is enabled on all three levels and if the SDS actually contains RFCache devices
- NVMe disks can't be used with RFCache as an acceleration device

NVRAM Acceleration Pools features and requirements:

- Must reside on NVDIMM devices
- The NVDIMM must be mapped to a DAX Device
- NVRAM Acceleration Pools cannot be disabled
- Migration to a different NVRAM pool is not supported

Limitation: NVDIMM failure on SDS node causes failure of all node's SSDs in the FG storage pool.

Data layout

VxFlex OS offers a space efficient store layout with Fine Granularity (FG)

With FG, since space allocation occurs at 4KB, the data stored on the disk is significantly reduced. Space allocation for Medium Granularity (MG) occurs at 1MB.

For FG, you must have NVDIMM configured as a DAX device. After NVDIMM is all set up in your environment, you need to configure an NVDIMM Acceleration Pool (for more information on Acceleration Pool, see "Acceleration Pools") and then you can select it in the Storage Pool (for more information on Storage Pool, see "Storage Pools").

FG enables data compression which allows for faster reads and writes. Data-compression is not supported for Medium Granularity (MG). It supports thin provisioned, zero padded volumes. Zero padded is enabled by default for FG and can't be disabled. Zero padding is disabled for MG.

FG supports persistent checksum for data integrity. For more information on persistent checksum, see "Checksum protection".

Volumes

You must add and configure volumes before enabling applications to access them.

The adding and mapping volume process is necessary, as part of the getting started process, before applications can access the volumes. In addition, you may create additional volumes and map them as part of the maintenance of the virtualization layer.

You can configure the caching option when creating the volumes, or you can change the Read RAM Cache feature later. If you want to enable the caching feature, ensure that the feature is also enabled in the backend of the system, for the corresponding Storage Pool and SDSs. For more information, see "Change Read RAM Cache volume settings".

Define volume names according to the following rules:

- Contains less than 32 characters
- Contains only alphanumeric and punctuation characters
- Is unique within the object type

VxFlex OS objects are assigned a unique ID that can be used to identify the object in CLI commands. You can retrieve the ID via a query, or via the object's property sheet in the VxFlex OS GUI. It is highly recommended to give each volume a meaningful name associated with its operational role.

Naming

Giving meaningful names to VxFlex OS objects can simplify volume-related actions.

It is recommended to name all VxFlex OS objects with meaningful names. This will make it easier when defining volumes, associating them with applications, etc.

From the previous example, the Storage Pools can be named "Capacity_Storage" and "Performance_Storage," which allows you to identify the different tiers.

As for Protection Domains, one example would be separating the SDSs used by the finance department from those used by the engineering department. This segregation of different departments is very beneficial in many aspects (security being one of them). Thus, one might name the domains "Financial-PD" and "Engineering-PD."

The Fault Sets could be called "FS_Rack01" and "FS_Rack02."

Protection and load balancing

VxFlex OS maintains the user data in a RAID-1 mesh mirrored layout. Each piece of data is stored on two different Fault Units. The copies are distributed over the storage devices according to an algorithm that ensures uniform load of each fault unit by terms of capacity and expected network load. Rebuild and rebalance processes are fully automated, but are configurable.

Rebuild

VxFlex OS initiates a Rebuild process in response to failure. Forward Rebuild refers to creating a new copy of the data on another server. Backward Rebuild refers to re-synchronizing one of the copies.

When a failure occurs, such as on a server, device or network failure, VxFlex OS immediately initiates a process of protecting the data. This process is called Rebuild, and comes in two flavors:

- Forward rebuild is the process of creating another copy of the data on a new server. In this process, all the devices in the Storage Pool work together, in a many-to-many fashion, to create new copies of all the failed storage blocks. This method ensures an extremely fast rebuild.
- Backward rebuild is the process of re-synchronization of one of the copies. This is done by passing to the copy only changes made to the data while this copy was inaccessible. This process minimizes the amount of data transferred over the network during recovery.

VxFlex OS automatically selects the type of rebuild to perform. This implies that in some cases, more data will be transferred to minimize the time that the user data is not fully protected.

Rebuild throttling

The rebuild throttling policy determines the priority of rebuild I/Os versus application I/Os when accessing SDS devices. The possible rebuild throttling policies are no limit on rebuild I/Os, limit concurrent I/Os per SDS device, Favor application I/Os and Dynamic Bandwidth Throttling.

Rebuild throttling sets the rebuild priority policy for a Storage Pool. The policy determines the priority between the rebuild I/O and the application I/O when accessing SDS devices. Please note that application I/Os are continuously served.

Applying rebuild throttling will on one hand increase the time the system is exposed with a single copy of some of data, but on the other hand, will reduce the impact on the application. One has to make a decision and choose the right balance between the two.

The following possible priority policies may be applied:

- No Limit: No limit on rebuild I/Os.

  Any rebuild I/O is submitted to the device immediately, without further queuing. Please note that rebuild I/Os are relatively large and hence setting this policy will speed up the rebuild, but will have the maximal effect on the application I/O.
- Limit Concurrent I/O: Limit the number of concurrent rebuild I/Os per SDS device (default).

  The rebuild I/Os are limited to a predefined number of concurrent I/Os. Once the limit is reached, the next incoming rebuild I/O waits until the completion of a currently executed rebuild I/O. This will complete the Rebuild quickly for best reliability, however, there is a risk of host application impact.
- Favor Application I/O: Limit rebuild in both bandwidth and concurrent I/Os.

  The rebuild I/Os are limited both in bandwidth and in the amount of concurrent I/Os. As long as the number of concurrent rebuild I/Os, and the bandwidth they consume, do not exceed the predefined limits, rebuild I/Os will be served. Once either threshold is reached, the rebuild I/Os wait until both I/O and bandwidth are below their thresholds. For example, setting the value to "1" will guarantee the device will only have one concurrent rebuild IO at any given moment, which will ensure the application IOs only wait for 1 rebuild IO at worst case.

  This imposes bandwidth on top of the Limit Concurrent I/Os option, which is a prerequisite to using this policy.
- Dynamic Bandwidth Throttling: This policy is similar to Favor Application I/O, but extends the interval in which application I/Os are considered to be flowing by defining a minimal quiet period. This quiet period is defined as a certain interval in which no application I/Os occurred. Note that the limits on the rebuild bandwidth and concurrent I/Os are still imposed.
- Default Values:
  - The default policy for rebuild is: Limit Concurrent I/O
  - Rebuild concurrent I/O Limit: 1 concurrent I/O

> **Note:**
> Rebuild throttling affects the system's performance and should only be used by advanced users.

Rebalance

When VxFlex OS detects that user data is not balanced across devices in the Storage Pool, it initiates a process to restore the balance in which data copies are moved from the most utilized devices to the least utilized.

Rebalance is the process of moving one of the data copies to a different server. It occurs when VxFlex OS detects that the user data is not evenly balanced across the fault units in a Storage Pool. This can occur as a result of several conditions such as: SDS addition/removal, device addition/removal, or following a recovery from a failure. VxFlex OS will move copies of the data from the most utilized devices to the least utilized ones.

Both Rebuild and Rebalance compete with the application IO for the system resources. This includes network, CPU and disks. VxFlex OS provides a very rich set of parameters that can control this resource consumption. While the system is factory-tuned for balancing between speedy rebuild/rebalance and minimization of the effect on the application IO, the user has very fine-grain control over the rebuild and rebalance behavior.

Rebalance throttling

The rebalance throttling policy determines the priority of rebalance I/Os versus application I/Os when accessing SDS devices. The possible rebuild throttling policies are no limit on rebalance I/Os, limit concurrent I/Os per SDS device, Favor application I/Os and Dynamic Bandwidth Throttling.

Rebalance throttling sets the rebalance priority policy for a Storage Pool. The policy determines the priority between the rebalance I/O and the application IO when accessing SDS devices. Please note that application I/Os are continuously served. Rebalance, unlike rebuild, does not impact the system’s reliability and therefore reducing its impact is not risky.

> **Note:**
> Rebalance throttling affects the system's performance and should only be used by advanced users.

The following possible priority policies may be applied:

- No Limit: No limit on rebalance I/Os.

  Any rebalance I/O is submitted to the device immediately, without further queuing. Please note that rebalance I/Os are relatively large and hence setting this policy will speed up the rebalance, but will have the maximal effect on the application I/O.

- Limit Concurrent I/O: Limit the number of concurrent rebalance I/Os per SDS device.

  The rebalance I/Os are limited to a predefined number of concurrent I/Os. Once the limit is reached, the next incoming rebalance I/O waits until the completion of a currently executed rebalance I/O. For example, setting the value to "1" will guarantee that the device will only have one rebalance IO at any given moment, which will ensure that the application IOs only wait for 1 rebalance IO in the worst case.

- Favor Application I/O: Limit rebalance in both bandwidth and concurrent I/Os.

  The rebalance I/Os are limited both in bandwidth and in the amount of concurrent I/Os. As long as the number of concurrent rebalance I/Os, and the bandwidth they consume, do not exceed the predefined limits, rebalance I/Os will be served. Once either limiter is reached, the rebalance I/Os wait until such time that the limits are not met again.

  This imposes a bandwidth limit on top of the Limit Concurrent I/Os option.

- Dynamic Bandwidth Throttling: This policy is similar to Favor Application I/O, but extends the interval in which application I/Os are considered to be flowing by defining a minimal quiet period. This quiet period is defined as a certain interval in which no application I/Os occurred. Note that the limits on the rebalance bandwidth and concurrent I/Os are still imposed.
- Default Values:
  - The default policy for rebalance: Favor Application I/O
  - Rebalance concurrent I/O Limit: 1 concurrent I/O per SDS device
  - Rebalance bandwidth limit: 10240 KB/s

  For information on Instant Maintenance mode that uses rebuild and rebalance, see topic "Instant Maintenance mode".

Checksum protection

VxFlex OS calculates and validates the checksum value for the payload during transit to protect data-in-flight. Checksum protection is applied to all I/Os.

This feature addresses errors that change the payload during the transit through VxFlex OS. VxFlex OS protects data in-flight by calculating and validating the checksum value for the payload at both ends.

Note: The feature is off by default.

Note: The checksum feature may have a major impact on performance and availability. Contact Customer Support to verify if your use case is relevant.

- During write operations, the checksum is calculated when the SDC receives the write request from the application. This checksum is validated just before each SDS writes the data on the storage device.
- During read operations, the checksum is calculated when the data is read from the SDS device, and is validated by the SDC before the data returns to the application. If the validating end detects a discrepancy, it will initiate a retry. The checksum will be done in the granularity of a sector (1/2KB).
- Pools with Fine Granularity with or without compression, have persistent checksum by default. This can't be changed.
  Each I/O goes through compression, checksum is calculated before it is written to the disk. There are two types of checksum:
  - Fine Granularity layout saves checksum data before and after processing to guarantee data integrity (compressed or not)
  - There are also system checksums for metadata

This feature applies to all I/Os: Application, Rebuild, Rebalance, and Migrate. The checksum is also kept in RMcache (Read Memory Cache), protecting every block that is maintained in SDS memory against memory corruption. The checksum feature can be enabled at the Protection Domain level, and defined at the Storage Pool level. The feature is T10/DIF-ready.

Fine Granularity layout checksum implementation

The Fine Granularity data layout has a default checksum whether it is compressed or not.

ESXi vStorage APIs for Array Integration (VAAI)

ESX vStorage APIs for Array Integration (VAAI) is a feature introduced in ESXi/ESX 4.1 that provides hardware acceleration functionality. It allows the host to offload specific virtual machine and storage management operations to compliant storage hardware. With the storage hardware's

assistance, the host performs these operations faster, and consumes less CPU, memory, and storage fabric bandwidth.

VAAI uses these fundamental operations:

- Atomic Test & Set (ATS), which is used during creation and locking of files on the VMFS volume
- Clone Blocks/Full Copy/XCOPY, which is used to copy or migrate data within the same physical array
- Zero Blocks/Write Same, which is used to zero-out disk regions
- Thin Provisioning in ESXi 5.x and later hosts, which allows the ESXi host to tell the array when the space previously occupied by a virtual machine (whether it is deleted or migrated to another datastore) can be reclaimed on thin provisioned LUNs.
- Block Delete in ESXi 5.x and later hosts, which allows for space to be reclaimed using the SCSI UNMAP feature.

The VxFlex OS supported VAAI features are:

- Atomic Test & Set (ATS)
- Zero Blocks/Write Same
- Thin Provisioning in ESXi 5.x and later hosts
- Block Delete in ESXi 5.x and later hosts

The following output is an example of typical output:

```
esxcli storage core device vaai status get -d
eui.7dbf14034834bbe01bf7e55800000002
eui.7dbf14034834bbe01bf7e55800000002
VAAI Plugin Name:
ATS Status: supported
```

`Clone Status: unsupported` This means that Clone Block/Full Copy/Xcopy is not supported.

`Zero Status: supported` This means that write same is supported.

`Delete Status: supported` This means that UNMAP is supported.

Note: Thin provisioning is not shown in VAAI output.

Cache

VxFlex OS offers a number of caching options, for the purpose of enhancing system performance.

The following caching options are supported by VxFlex OS:

- RAM Read Cache (using DRAM server memory)
- Read Flash Cache (using SSD and NVMe SSD devices)

In addition, the following caching solutions are available:

- CacheCade (using SSD devices) - available in VxRack Node 100 Series systems
- DAS Cache (using SSD devices) - available in VxFlex Ready Node systems only

Note: For the most updated list, see the DAS Cache support matrix at https://topics-cdn.dell.com/pdf/dell-sandisk-das-cache_Reference-Guide_en-us.pdf.

SSDs used for caching cannot be used for storage purposes.

Note: If CacheCade or DAS Cache is used, you must configure External acceleration. Some devices are already benefitting from an external (For example, a non-VxFlex OS related)

acceleration/caching mechanism. This input is required in order to avoid alerts about media type mismatches. For example, when the user adds an HDD device which the SDS perceives as being too fast to fit the HDD criteria.

The following table summarizes information about the caching modes provided by the system.

Table 2 Caching modes

Mode	Description	Considerations	Default Setting
RAM Read Cache (RMcache)	Read-only caching performed by server RAM.	RAM Read cache, the fastest type of caching, uses RAM that is allocated for caching. Its size is limited to the amount of allocated RAM. **Note:** The amount that may be allocated is limited, and can never be the maximum available RAM.	Disabled, except when storage-only nodes are deployed.
Read Flash Cache (RFcache)	Read-only caching performed by one or more dedicated SSD or NVMe SSD devices in the server.	RFcache uses the full capacity of SSD devices (up to eight) to provide a larger footprint of read-only LRU (Least Recently Used) based-caching resources for the SDS. This type of caching reacts quickly to workload changes to speed up HDD Read performance. Several SSD devices can be allocated to a shared cache pool, and therefore the cache size is limited in size only by the amount of SSDs allocated for this purpose. The RFcache driver must be installed during deployment. Caching devices can be defined either during the installation process or after deployment. Limitations: Windows backend is not supported. Support matrix: • An RFcache device (flash device) can be partitioned only on Linux. • An SDS storage/source device cannot be partitioned if it needs to be accelerated by RFcache. • An SDS storage/source device as a file (over file system), cannot be accelerated by RFcache. • CloudLink Data at Rest Encryption (D@RE) is not supported on devices used for the RFcache feature.	Disabled
CacheCade	Read and write-back caching performed by one or more	CacheCade uses the full capacity of one or more SSD devices to provide a large footprint of both read and write-back	Disabled

Table 2 Caching modes (continued)

Mode	Description	Considerations	Default Setting
	dedicated SSD devices in the server.	caching resources to the SDS. This caching mode moves "hot" (active) chunks of data from HDDs to cache, for Read and Write buffering. For write-back caching, the write is temporarily written to the SSD, which is much faster than an HDD, allowing faster response of the SDS to write acknowledgment. Two SSD devices can be allocated to a shared cache pool, up to a maximum size of 512 GB in total. **Note:** If a fault occurs in the caching device before the writes have been offloaded, all the HDD devices cached by CacheCade acquire failed status, and a rebuild process commences in VxRack Node 100 Series. Once the rebuild is over, the caching disk can be replaced, all caching has stopped in the storage pool, and the HDD members in the storage pool can be cleared of errors.	
DAS Cache	Read and write-back caching performed by one or more dedicated SSD devices in the server	DAS Cache uses the full capacity of one or more SSD devices to provide a large footprint of both read and write-back caching resources to the SDS. This caching mode moves "hot" (active) chunks of data from HDDs to cache, for Read and Write buffering. For write-back caching, the write is temporarily written to the SSD, which is much faster than an HDD, allowing faster response of the SDS to write acknowledgment. One SSD device can accelerate several HDDs (in DAS Cache they are called "Volumes"). Striping the Cache on two devices is not supported in the VxFlex Ready Node solution. CloudLink Data at Rest Encryption (D@RE) is not supported on devices used for the DAS Cache feature. **Note:** If a fault occurs in the caching device before the writes have been offloaded, all the HDD devices cached by DAS Cache acquire failed status, and a rebuild process commences in VxFlex OS. Once the rebuild is over, the caching disk can be replaced, all caching has stopped in the storage pool, and the HDD members in the storage pool can be cleared of errors.	Disabled

Table 2 Caching modes (continued)

Mode	Description	Considerations	Default Setting
XCache	XCache accelerates the performance of PHP on servers	• XCache device (flash device) can be partitioned only over Linux • SDS device can't be partitioned at all if you want to accelerate it by the XCache module • SDS device as a file (over file system) also can't be accelerated by the XCache module • If you are running Linux and try to accelerate an SDS device by a partitioned flash device, it should work	

The following table illustrates the caching support matrix:

Table 3 Caching support matrix

System	RFcache	RMcache	DAS Cache	CacheCade
VxFlex OS	Yes	Yes		
VxFlex Ready Node PowerEdge 13G servers	Yes	Yes	Yes	
VxFlex Ready Node PowerEdge 14G servers	Yes	Yes		
VxRack Node 100 Series	Yes	Yes		Yes

Networking

This section describes the various considerations for choosing how to manage inter-node communication - via a separate network with access to the VxFlex OS components or on the same network.

In VxFlex OS, inter-node communication (for the purposes of managing data locations, rebuild and rebalance, and for application access to stored data) can be done on one IP network, or on separate IP networks. Management (via any of the management interfaces) can be done in the following ways:

- Via a separate network with access to the other VxFlex OS components
- On the same network

These options can be configured a) during deployment in the full VxFlex OS Installer (via the CSV topology file) and using the VMware plug-in, as well as b) after deployment with the CLI.

This section describes how to choose from these options, depending on your organization's requirements, security considerations, performance needs, and IT environment.

VxFlex OS networking considerations:

- Single IP network: All communications and IOs used for management and for data storage are performed on the same IP network. This setup offers the following benefits:
  - Ease of use

- Fewer IP addresses required

- Multiple separate IP networks: Separate networks are used for management and for data storage, or separate networks are used within the data storage part of the system. This setup offers the following benefits:
  - Security
  - Redundancy
  - Performance
  - Separate IP roles in order to separate between customer data and internal management

    Note: Network high availability can be implemented by using NIC-bonding (refer to https://www.dellemc.com/collateral/white-papers/h17332-dellemc-vxflex-os-networking-best-practices.pdf) or by using several data networks in VxFlex OS.

  For more information about MTU performance considerations and best practices, see the *VxFlex OS Configure and Customize Guide*.

  Note: The MDM cluster IP address limit is 16 IP addresses, which includes all cluster members (Master, Slave, Standby Master, and Standby Slaves).

  The following table describes the range of potential IP address configurations:

Table 4 IP address configurations in VxFlex OS (based on CSV file)

Column in CSV file	MDM Mgmt IP	MDM IPs	SDS All IPs	SDS-SDS Only IPs	SDS-SDC Only IPs
Comments	Management Access	Control Network	Rebuild and Data Path Network	Rebuild Network	Data Path Network
	Optional, but recommended; not applicable for Tie Breaker IP addresses that can be used to provide access to VxFlex OS management applications, such as VxFlex OS GUI, CLI, REST API, OpenStack. This IP address must be externally accessible.	Mandatory IP addresses used for MDM control communications with SDSs and SDCs, used to convey data migration decisions, but no user data passes through the MDM. Must be on the same network as the data network. Must be externally accessible if no MDM Management IP addresses are used. MDM Virtual IP is supported on the data network.	IP addresses used for both SDS-SDS and SDS-SDC communications. These IP addresses will also be used to communicate with the MDM	IP addresses used for SDS-SDS communication only. These addresses are used for rebuild & rebalance operations. These IP addresses will also be used to communicate with the MDM.	IP addresses used for SDS-SDC communication. These addresses are only used for read-write user data operations.

The following combinations can be used for SDS/SDC:

- Only *SDS All IPs*
- Only *SDS-SDS Only IPs* + *SDS-SDC Only IPs*
- *SDS All IPs* + either *SDS-SDS Only IPs* or *SDS-SDC Only IPs* (can be used in cases of multiple networks; ensure that you do not use the same IP address more than once in the networks).

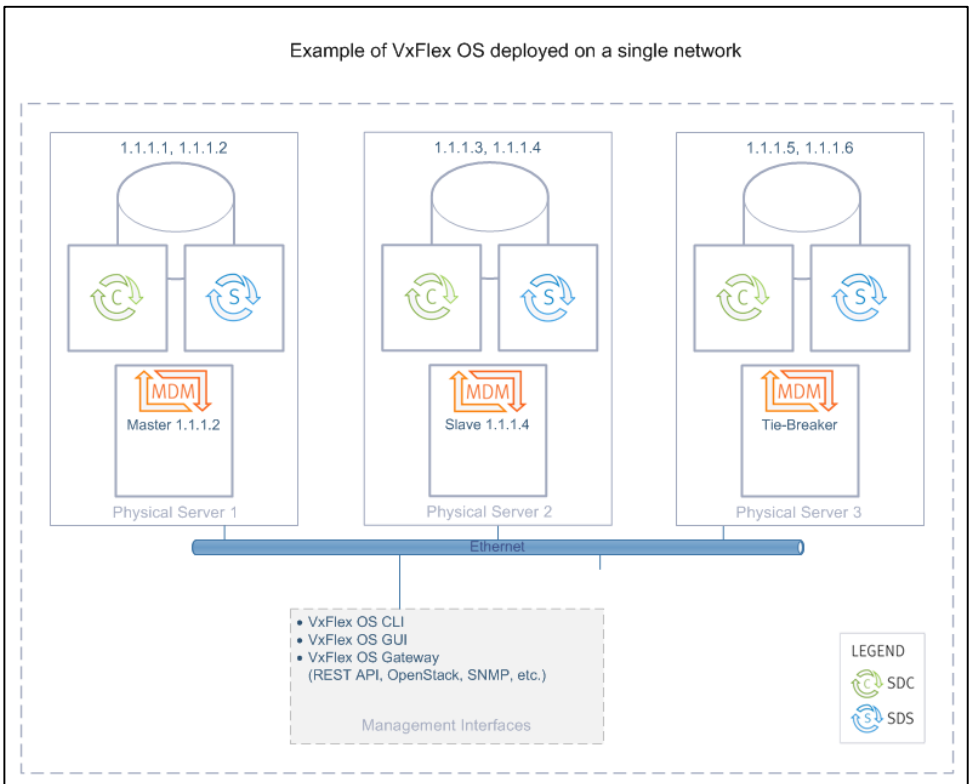
- *SDS All IPs* + both *SDS-SDS Only IPs* and *SDS-SDC Only IPs* (can be used in cases of multiple networks; ensure that you do not use the same IP address more than once in the networks).

  Note:

  In the following example drawing for separate networks, a very simple example is shown, where the management and storage parts of the system are on different networks. In more complex configurations, MDMs, SDCs and SDSs can be on separate networks. Up to 8 separate networks per VxFlex OS installation are supported.

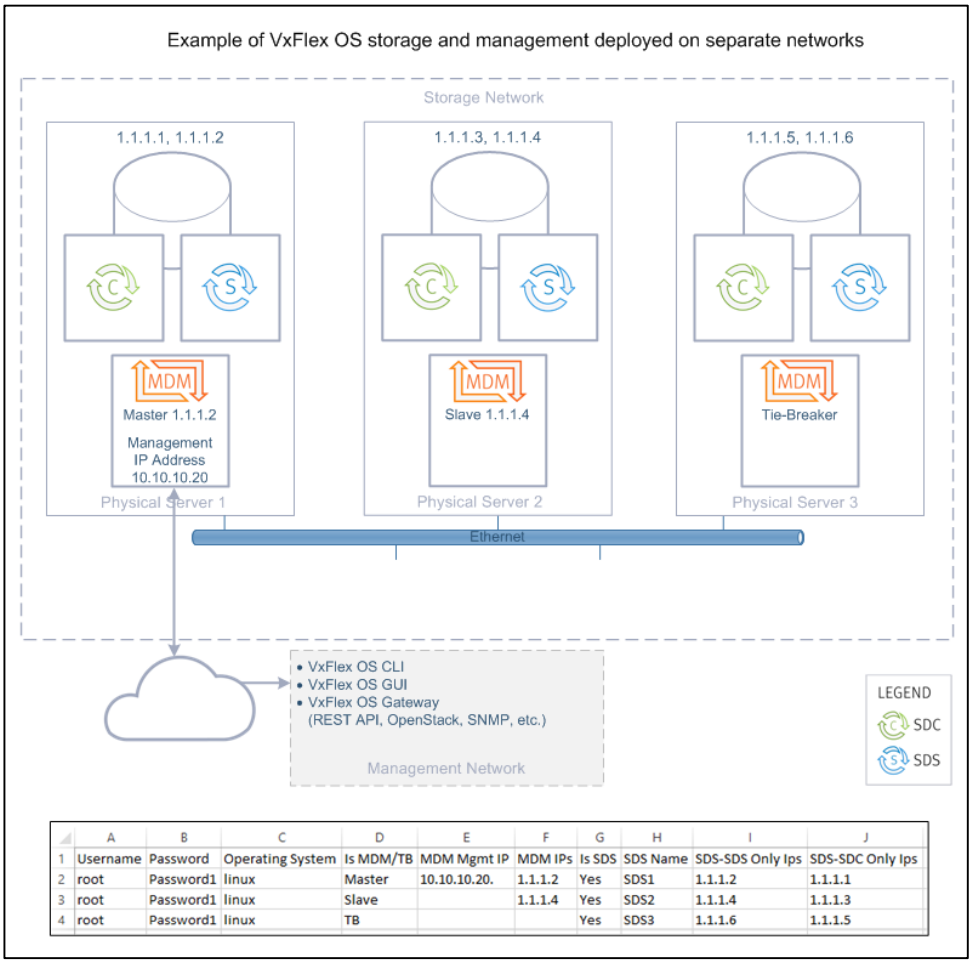
The following figures show example configurations and the corresponding fields in a CSV configuration file:

Figure 5 VxFlex OS deployed on a single network



	A	B	C	D	E	F	G	H	I	J
1	Username	Password	Operating System	Is MDM/TB	MDM Mgmt IP	MDM IPs	Is SDS	SDS Name	SDS-SDS Only Ips	SDS-SDC Only Ips
2	root	Password1	linux	Master		1.1.1.2	Yes	SDS1	1.1.1.2	1.1.1.1
3	root	Password1	linux	Slave		1.1.1.4	Yes	SDS2	1.1.1.4	1.1.1.3
4	root	Password1	linux	TB			Yes	SDS3	1.1.1.6	1.1.1.5

Figure 6 VxFlex OS deployed on separate networks



	A	B	C	D	E	F	G	H	I	J
1	Username	Password	Operating System	Is MDM/TB	MDM Mgmt IP	MDM IPs	Is SDS	SDS Name	SDS-SDS Only Ips	SDS-SDC Only Ips
2	root	Password1	linux	Master	10.10.10.20.	1.1.1.2	Yes	SDS1	1.1.1.2	1.1.1.1
3	root	Password1	linux	Slave		1.1.1.4	Yes	SDS2	1.1.1.4	1.1.1.3
4	root	Password1	linux	TB			Yes	SDS3	1.1.1.6	1.1.1.5

VMware limitation

When deployed on VMware, VxFlex OS, multiple IP subnets cannot be on the same subnets.

Multiple IP subnets used for the VxFlex OS data network cannot be on the same subnet in a VMware setup.

For more information, see the VMware limitation in the following link:

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2010877

VxFlex OS only supports the following network configurations when deployed on VMware:

- A single data storage network
- Two or more data networks, each on separate IP subnets
- A single IP data network using several NIC-bonding configurations, or vSwitch load balancing

Virtual IP Address

Virtual IP addresses can be defined for the MDM cluster.

Up to four virtual IP addresses can be defined for the MDM cluster. SDCs are then mapped to the MDM cluster's virtual IP addresses, instead of to static MDM IP addresses. MDMs are sometimes switched during normal operation of the cluster, and the virtual IP address will always be mapped

to the active MDM. The use of virtual IP addresses simplifies maintenance procedures on the MDM cluster, because system components communicate via the virtual IP addresses. Therefore, SDCs do not need to be reconfigured when a server hosting an MDM is replaced.

In new installations in Linux environments, the MDM cluster's virtual IP address can be added and mapped using the VxFlex OS Installer CSV file. In VMware environments, virtual IP addresses are mandatory, and configuration is performed using the VxFlex OS VMware Installation Wizard, in the Configure SVM stage. The REST API can also be used to add virtual IP addresses to the cluster. In all cases, a virtual IP NIC placeholder must be mapped to each virtual IP address. Ensure that there are NICs available for this purpose.

Existing systems may be extended to include additional MDMs to a cluster. The new MDMs should be mapped to the existing virtual IP addresses.

If virtual IP addresses need to be modified, you must use the CLI or the REST API (not the VxFlex OS Installer or the vSphere plug-in), and it must be done with extreme caution.

All SDCs will require reconfiguration, to reflect the changes made to the MDM cluster. Otherwise, the SDCs will not be able to communicate with the MDM cluster, and volumes will not be accessible.

Monitor of SDC and SDS connections

The system monitors all connections between SDCs and SDSs and sends out an alert when an active connection between an SDC and an SDS goes down.

To effectively monitor SDC and SDS connections, the MDM collects connectivity updates from all of the SDCs. The MDM posts events whenever an SDC connects to or disconnects from a specific SDS IP address. The MDM frequently analyzes the connectivity status to determine the current system state. The system does not send out alerts for temporary connectivity issues that are resolved in less than 10 seconds.

The following are the possible connectivity states between SDCs and SDSs in the system:

- All connected
- One SDC is disconnected from one SDS
- One SDC is disconnected from one SDS IP address
- One SDC is disconnected from all SDSs
- All SDCs are disconnected from one SDS
- All SDCs are disconnected from one SDS IP address
- All SDCs are disconnected from all SDSs
- Multiple disconnections

When the system's connectivity state changes to any state other than `All Connected`, an alert is displayed in the VxFlex OS GUI and is written to the MDM event log. Once an alert is generated, you can use the SCLI to query details on the disconnection using the command `scli --query_sdc_to_sds_disconnections`. For more information about running SCLI commands, see the *CLI Reference Guide*.

The MDM does not monitor the connectivity state of SDCs or SDSs in the following scenarios:

- SDS is in maintenance mode
- SDS is disconnected from the MDM
- SDS is in the process of being removed
- SDC is disconnected from the MDM for more than two minutes
- SDC is not approved

The alerts are detailed in "Alerts in SNMP, VxFlex OS GUI, REST, and ESRS" in the *Monitoring VxFlex OS* .

S.M.A.R.T. hardware monitoring

The VxFlex OS bare-metal solution now provides monitoring capabilities for RAID controllers and storage devices compatible with S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology) protocols.

In Linux-based environments, S.M.A.R.T.-compatible HDDs, SSDs and RAID storage controllers can be monitored for S.M.A.R.T. attributes such as temperature, SSD wear level, and error counters. LEDs can also be lit on these hardware devices, to simplify physical identification for maintenance purposes.

Each hardware vendor defines specific thresholds for the S.M.A.R.T. attributes. This feature currently supports storage devices controlled by LSI, HP and Dell RAID controllers, and stand-alone devices. During system deployment, an external monitoring tool is installed as part of the LIA on each node. Additional RAID controller tools must be installed manually after system deployment: storcli for LSI RAID controllers, hpssacli for HP RAID controllers, or perccli for DELL RAID controllers. These tools are used by the system to collect the counters that are returned to the MDM.

Note: In some cases, LSI RAID controllers may report vendor information as "AVAGO" instead of LSI.

The MDM queries the SDSs at set intervals, and stores the returned information. This information can be viewed using CLI queries. In addition, when thresholds are crossed for S.M.A.R.T. attributes, alerts are generated by the system.

When the CLI is used to query device information, physical device information, such as serial number, model name, vendor etc., temperature, and wear level information (for SSDs only) is included in the returned response.

For information about the use of CLI commands, see the *VxFlex OS CLI Reference Guide*.

For information about the use of REST API URIs, see the *VxFlex OS User Guide*.

You can use the VxFlex OS GUI to monitor S.M.A.R.T.-related alerts in the **Alerts** view.

In addition, SNMP traps and ESRS alert codes can be used to monitor alerts triggered by devices compatible with S.M.A.R.T.

List of approved RAID controllers

Provides high-level specifications of RAID controllers, which are tested and certified by VxFlex OS.

VxFlex OS-certified RAID controllers

The following table describes the VxFlex OS-certified RAID controllers:

Manufacturer	Specifications
HP	• Model Name: Smart Array P440ar • Vendor Name: HP • Firmware Version: 3.56 • Driver Version: 3.4.10 • Driver Name: hpsa

Manufacturer	Specifications
DELL	• Model Name: PERC H730 Mini • Vendor Name: Dell • Firmware Version: 25.5.5.0005 • Driver Version: 07.700.52.00 and above • Driver Name: megaraid_sas
LSI	• Model Name: LSI MegaRAID SAS 9271-8i • Vendor Name: LSI • Firmware Version: 23.12.0-0021 • Driver Version: 06.810.09.00-rh • Driver Name: megaraid_sas
	• Model Name: LSI MegaRAID SAS 9271-8i • Vendor Name: LSI • Firmware Version: 23.12.0-0018 • Driver Version: 06.805.06.01-rc • Driver Name: megaraid_sas
	• Model Name: LSI MegaRAID SAS 9271-8i • Vendor Name: LSI • Firmware Version: 23.12.0-0021 • Driver Version: 06.805.06.01-rc • Driver Name: megaraid_sas
	• Model Name: LSI MegaRAID SAS 9271-8i • Vendor Name: LSI • Firmware Version: 23.34.0-0005 • Driver Version: 06.810.08.00 • Driver Name: megaraid_sas

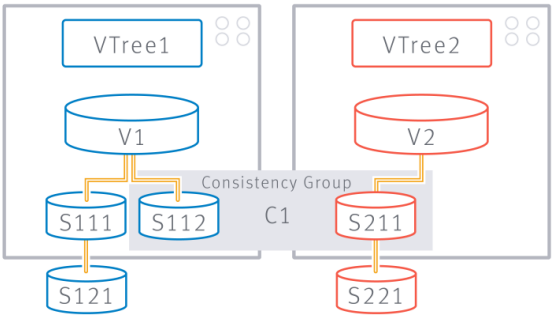
Snapshots

VxFlex OS enables you to create instantaneous copies of volumes, i.e. snapshots. A snapshot can be manipulated as any volume exposed by the storage system. Snapshots taken at the same time form a consistency group and can be manipulated as a group.

The VxFlex OS storage system enables you to take snapshots of existing volumes, up to 128 per volume. The snapshots are thin provisioned and are extremely quick. For more information about thin provisioning, see "SAN virtualization layer".

Once a snapshot is generated, it becomes a new, unmapped volume in the system. You can manipulate it in the same manner as any other volume exposed by the VxFlex OS storage system.

Figure 7 Snapshot operations



The structure related to all the snapshots resulting from one volume is referred to as a V-Tree (short for volume tree). When taking a snapshot in the system, you can specify more than one volume. All snapshots taken together form a consistency group. They are consistent in the sense that they were all taken at the same time. So if there is a contextual relationship between the data contained by all the snapshot members, then that set is meaningful. The consistency group allows manipulation of the entire set.

If you remove an entire consistency group, all of the snapshots that were taken together will be removed. In RED, S211 is a snapshot of V2. Since S112 and S211 were taken together, they compose a consistency group designated as C1.

Note: The consistency group is only for convenience purposes. There are no protection measures done by VxFlex OS to conserve the consistency group. For example, you can remove a snapshot that is a member of a consistency group.

Note: Support of removal of a parent snapshot in the system. You can merge the parent to a child snapshot.

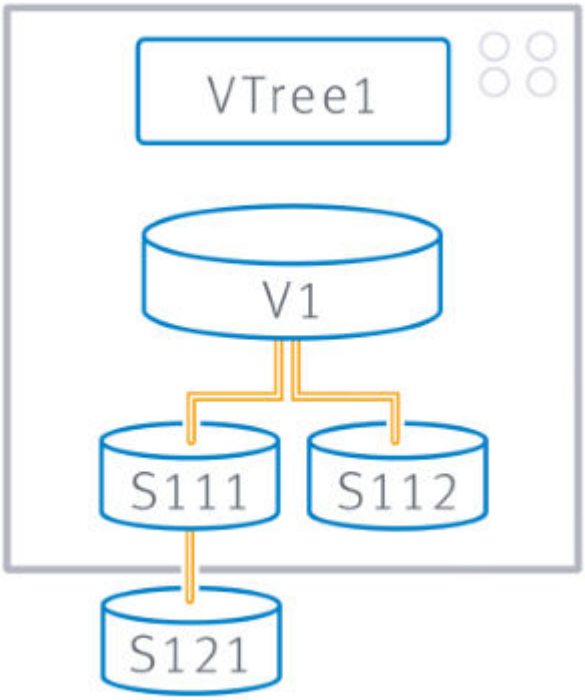
Note: It is recommended not to trim volumes that contain snapshots as the logical capacity may not shrink.

V-Trees

A V-Tree consists of the root volume ans all its descendant volumes and snapshots.

A V-Tree (short for volume tree) is the structure comprised of a volume and the snapshots resulting from that volume. It is a tree spanning from the source volume at its root, whose descendants are either snapshots of the volume itself or snapshots of a snapshot. In the V-Tree diagram, S_{111} and S_{112} are snapshots of V_1. S_{121} is a snapshot of snapshot S_{111}. Together, V_1 and S_{1xy} are the V-Tree of V_1.

Figure 8 V-Tree diagram



Snapshot Policies

Snapshot Policies enable you to define policies where you can configure the number of snapshots to take at a given time for one or more volumes.

The snapshots are taken according to the rule defined. You can define the time interval in-between two rounds of snapshots as well as the number of snapshots to retain, in a multi-level structure. For example take snapshots every x minutes/hours/days/weeks. There are one to six levels, with the first level having the most frequent snapshots.

Example:

Rule: Take snapshots every 60 minutes

Retention Levels:

- 24 snapshots
- 7 snapshots
- 4 snapshots

After defining the parameters, you must then select the source volume to add to the Snapshot Policy. You can add multiple source volumes to a Snapshot Policy, but a specific volume can only be the source volume of a single policy. Only one volume per VTree may be used as a source volume of a policy (any policy). For more information on V-Trees, see "Snapshots" in the *Getting to Know Guide*.

When you remove the source volume from the policy, you must choose how to handle auto-snapshots. Snapshots created by the policy are referred to as auto-snapshots. Your selection depends on if there are locked auto snapshots.

- If the source volume has no auto snapshots, it doesn't matter if you select **Detach auto snapshots** or **Remove auto snapshots**.

- If the source volume has auto snapshots but none of them are locked, you can choose to detach all snapshots. They become regular snapshots as if the user created them manually. If you select **Remove auto snapshots**, they are deleted.
- If the source volume has locked auto snapshots, you can choose to detach all snapshots. They become regular snapshots, as if the user created them manually. If you remove them, those that are not locked are removed, while the auto snapshots which are locked are detached.

Performance

System performance can be enhanced by implementing the high performance profile. This is the default profile for all system objects (MDMs, SDSs, SDCs, and SPs).

The profile of an SDS can't be changed to compact if it has at least on Fine Granularity device

The MDM, SDC, and SDS components are deployed with the default performance profile. To enhance system performance, you can configure each component with the high profile, either during or after deployment.

The high performance profile should be enabled on servers equipped with processors based on Haswell-equivalent or newer architectures and 48 GB of RAM. Other configurations may benefit from this setting as well.

To configure the performance profile during deployment, see the "Configure performance profile during deployment section" of the *VxFlex OS Deployment Guide*.

To configure the performance after deployment, use the VxFlex OS GUI, as described in the *VxFlex OS Configure and Customize Guide User Guide*. You can also use the `set_performance_parameters` command, as described in the *VxFlex OS CLI Reference Guide*.

For more information on the difference between the default and the high performance profile, see the *VxFlex OS Configure and Customize Guide* .

Migrate volumes

You can migrate volumes to a different Storage Pool or Protection Domain within your VxFlex OS system.

You can migrate volumes from the following storage layers:

- From one Storage Pool to another within the same Protection Domain
- From one Storage Pool to another across different Protection Domains
- Volume Tree and all it's snapshots are migrated together
- From thick to thin
- From Fine/Medium to Medium/Fine Granularity

Requirements to migrate volumes:

- Zero padding of pools - especially moving from Medium to Fine Granularity for compression benefits
- Medium Granularity pools should have zero padding enabled

Other functions

This section lists VxFlex OS additional functions.

VxFlex OS includes the following functions:

- EMC Secure Remote Support (ESRS)

  ESRS support enables secure, high-speed, 24x7, remote connection between EMC and customer installations, including:

  - Remote monitoring
  - Remote diagnosis and repair
  - Daily sending of system events (rsyslog output), alerts, and VxFlex OS topology
- Syslog

  The MDM syslog service can send events, via TCP/IP, to RFC 6587-compliant remote (or local) Syslog servers. Messages are sent with facility local0, by default. Once the syslog service is started, all events will be sent until the service is stopped.
- Get Info

  Get Info assembles a ZIP file of system logs for troubleshooting. You can run this function from a local node for its own logs, using the CLI, or by using the VxFlex OS Installer to assemble logs from all MDM and SDS nodes in the system. In addition to the log files, a visual snapshot of the VxFlex OS GUI, from the time you perform the operation, can be saved, to better enable support options.

  The Get Info function is described in the *Log Collection Guide*.
- Quality of Service (QoS)

  You can adjust the amount of bandwidth and storage that any given SDC can use. You can configure this with the CLI and the REST interface, on a per client/per volume basis.
- Running Remove script on system via Gateway

  In VxFlex OS Gateway, you can run a user provided script as part of the Non-disruptive orchestration process (like NDU), which is usually intended for OS patching. This is only supported on Linux, specifically: RHEL and SLES.

  Using this feature includes two main steps:

  1. To manually copy the script file to each VxFlex OS host using these script names:
     - patch_script
     - verification_script
  2. Copy the script to the `~/lia/bin` folder and add execution permissions.
     RC codes are saved in the LIA log and an error is returned if needed to the Gateway.
  3. Execute the scripts from VxFlex OS Gateway.

  Note: You need to test the patch_script and verification_script before running the process on the VxFlex OS Gateway
- Background Device Scanner

  The Background Device Scanner ("scanner") enhances the resilience of VxFlex OS by constantly searching for, and fixing, device errors before they can affect your system. This provides increased data reliability than the media's checksum scheme provides. The scanner seeks out corrupted sectors of the devices in that pool, provides SNMP reporting about errors found, and keeps statistics about its operation.

  When a scan is completed, the process starts again, thus adding constant protection to your system.

  For Fine Granularity (FG), the Background Device Scanner is enabled by default. It goes through a cycle of each SSD and compares the physical checksums against the data in the Logs and Metadata. The Bandwith Limit is set to 1024 KBs by default.

  You can set the scan rate (default: 1 MB/second per device), which limits the bandwidth allowed for scanning, and choose from the following scan modes:

- Device only mode

  The scanner uses the device's internal checksum mechanism to validate the primary and secondary data. If a read succeeds in both devices, no action is taken. If a faulty area is read, an error will be generated.

  If a read fails on one device, the scanner attempts to correct the faulty device with the data from the good device. If the fix succeeds, the error-fixes counter is increased. If the fix fails, a device error is issued.

  Note:
  A similar algorithm is performed every time an application read fails on the primary device.

  If the read fails on both devices, the scanner skips to the next storage block.

- Data comparison mode (only available if zero padding is enabled)

  The scanner performs the same algorithm as above, with the following additions:

  After successful reads of the primary and secondary copies of the data, the scanner calculates and compares their checksums. If this comparison fails, the compare errors counter is increased, and the scanner attempts to overwrite the secondary device with the data from the primary device. If this fails, a device error is issued.

  The scanning function is enabled and disabled (default) at the Storage Pool level, and this setting affects all devices in the Storage Pool. You can make these changes at anytime, and you can add/remove volumes and devices while the scanner is enabled.

  When adding a device to a Storage Pool in which the scanner is enabled, the scanning will start about 30 seconds after the device is added.

- AD over LDAP or LDAPS authentication

  User authentication may be done using AD (Active Directory) over LDAP (Lightweight Directory Access Protocol) or LDAPS (Secure LDAP). VxFlex OS can support both AD users that are fully controlled through the customer's existing centralized location, and local users (as has been supported in earlier VxFlex OS versions). You can associate groups from the AD with the existing VxFlex OS roles in order to ensure the Role-Based Access (RBAC) model. When a user logs on to VxFlex OS, the MDM identifies that the user belongs to the AD domain, and authenticates the user against the AD server over secured communications. Once the user is authenticated, VxFlex OS accepts the group to which the user belongs according to the AD, and associates the appropriate role and its user permissions to that user. The AD implementation is fully redundant.

  Note:
  The authorization permissions of each role are defined differently for local authentication, and for LDAP/LDAPS authentication.

  The benefits of using AD over LDAP/LDAPS include:

  - Full control of VxFlex OS users through the main user repository
  - No need to specify a local user for each customer

    If the AD directory is down, the administrator can always use local users to maintain VxFlex OS.

- LDAP for LIA

  You can deploy a system with a LIA user already configured to use LDAP. After upgrade, you can switch LIA from local user to LDAP user. You can add\remove up to eight LDAP servers.

- Multiple LDAP support in the Gateway
  For post deploy, use the FOSGWTool to add LDAP servers to the Gateway login authority. You can add\remove up to eight LDAP servers.

- Oscillating failure handling

  The Oscillating Failures feature detects and reports various oscillating failures, in cases when components fail repeatedly and cause unnecessary failovers, and therefore disruptions to normal system operation. Typical examples of oscillating failures include:

  - A disk that accepts some I/Os and rejects others
  - A node with interrupted connectivity
  - A node that is constantly busy and therefore handles some I/Os too slowly
  - A disk that is sometimes slow to respond
  - A network that is experiencing disruptions

    The smart detection of such failures provides the ability to handle error situations, and to reduce their impact on normal system operation. Oscillating failure handling can be set for MDMs, SDSs and for SDCs. For SDSs, failure handling can be defined per Protection Domain or per Storage Pool.

  - Oscillating failure counters

    The following table describes the oscillating failure counters:

Oscillating failure counters	Description
(sds_sds/sdc_mdm/sdc_sds/mdm_sds) network_disconnections	Measures the number of network disconnections (socket closed) between two components per IP address
sds_decoupled	Measures the number of times an SDS process is down, as detected by the MDM
sds_configuration_failures	Measures the number of times the MDM fails to configure an SDS, when connecting to an SDS (failures occur during the reconfiguration phase)
sds_receive_buffer_allocation_failures	Measures the number of times an SDS fails to allocate buffer for receiving messages
sdc_long_operations	Measures the number of SDC RPC operations that take longer than the predefined threshold (default threshold is 5 seconds)
sdc_memory_allocation_failures	Measures the number of memory allocation failures in each SDC
sdc_socket_allocation_failures	Measures the number of socket allocation failures in each SDC
sds_device_long_successful_ios	Measures the number of successful IOs to an SDS device, which take longer than the predefined threshold (default threshold is 250 milliseconds)

- Secure connectivity with external components

  This feature allows external components to authenticate the MDM. After authentication, communication between the MDM and external components is performed using TLS (Transport Layer Security) protocols.

  Secure communication with the MDM is authenticated by the following components:

- CLI client

  Note:
  If the secure mode is not enabled, modifications are necessary to run SCLI commands.
- VxFlex OS Gateway
- VxFlex OS GUI client
- VxFlex OS Installer client

  Once added in the trust point, all communications will require authentication, followed by communications over TLS. The same method is employed between the VxFlex OS Installer and all LIAs.

Implementing VxFlex OS

This section outlines the steps for implementing VxFlex OS.

Implementing VxFlex OS is, in general, a two-step process: first build the physical storage layer, then configure the virtual SAN layer on top of it.

Physical layer

This section describes the VxFlex OS physical layer and outlines the steps for implementing it.

The physical layer consists of the hardware (servers with storage devices and the network between them) and the VxFlex OS software installed on them.

Typically, each SDS is physically located on a separate server.

To implement the physical layer, perform the following steps:

1. Install the MDM component on the MDM nodes in one of the following configurations:
   - Three-node redundant cluster (one Master MDM, one Slave MDM, and one Tie Breaker).
   - Five-node redundant cluster (one Master MDM, two Slave MDMs, and two Tie Breakers).
   - Single node (one master MDM).

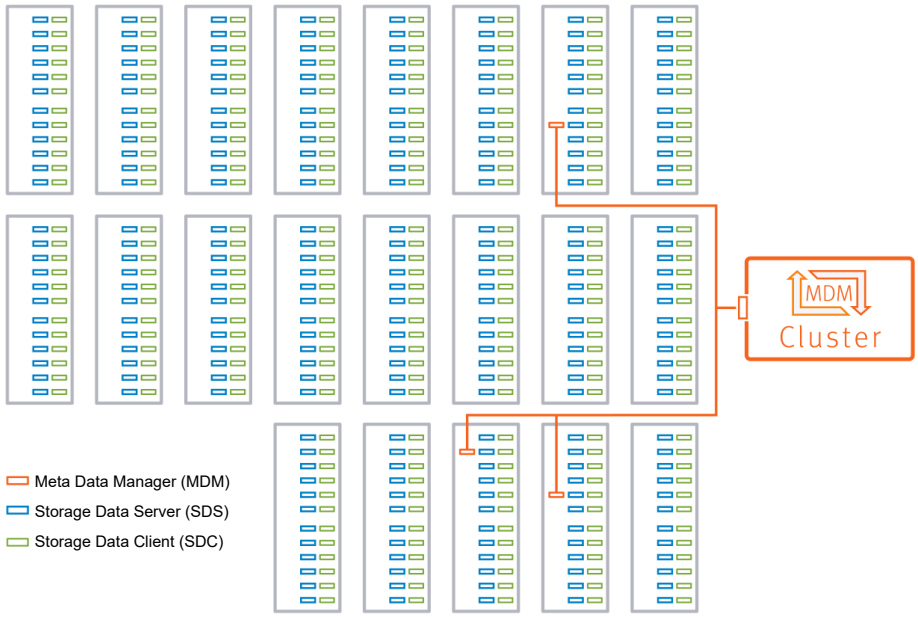
     NOTICE
     It is not recommended to use Single Mode in production systems, except in temporary situations. The MDM contains all the metadata required for system operation. Single Mode has no protection, and exposes the system to a single point of failure.

   MDMs do not require dedicated nodes. They can be installed on nodes hosting other VxFlex OS components.
2. Install the SDS component on all nodes that will contribute some, or all, of their physical storage.

   Divide the SDS nodes into Protection Domains. Each SDS can be a member of only one Protection Domain.

   Per Protection Domain, divide the physical storage units into Storage Pools, and optionally, into Fault Sets.
3. Install the SDC component on all nodes on which the application will access the data exposed by the VxFlex OS volumes.

Figure 9 Physical layout example—3-node cluster



Communication is done over the existing LAN using standard TCP/IP. The MDM and SDS nodes can be assigned up to eight IP addresses, enabling wider bandwidth and better I/O performance and redundancy.

You can perform physical layer setup using the following methods:

- VxFlex OS Installer, a web-client based tool
- VxFlex OS VMware plug-in a VMware plug-in
- Manual installation procedures

  After completing this installation, the physical layer is ready, and can expose a virtual storage layer.

SAN virtualization layer

This section details the steps for exposing the virtual SAN devices to servers with installed and confiured SDCs.

The MDM cluster manages the entire system. It aggregates the entire storage exposed to it by all the SDSs to generate a virtual layer - virtual SAN storage. Volumes can now be defined over the Storage Pools and can be exposed to the applications as a local storage device using the SDCs.

To expose the virtual SAN devices to your servers (the ones on which you installed and configured SDCs), perform the following:

1. Define volumes. Each volume defined over a Storage Pool is evenly distributed over all members using a RAID protection scheme. By having all SDS members of the Storage Pool participate, VxFlex OS ensures:

   - Highest and most stable and consistent performance possible
   - Rapid recovery and redistribution of data
   - Massive IOPS and throughput

   You can define volumes as follows:

   - Thick
     Capacity is allocated immediately, even if not actually used. This can cause capacity to be allocated, but never used, leading to wasted capacity.

Thick capacity provisioning is limited to available capacity.

- Thin
  Capacity is "on reserve," but not allocated until actually used. This policy enables more flexibility in provisioning.

  Whereas thick capacity is limited to available capacity, thin capacity provisioning can be oversubscribed, as follows:

  Maximum thin capacity provisioning = 5 * (gross capacity - used capacity)

  When capacity usage reaches the level where it may cause IO errors, alerts are generated. At certain higher capacity levels, volumes (even thin volumes) can no longer be created.

  Example:

  In a system with 3 SDSs, each with 10 TB, there are 30 TB of storage.

  In the system, there is already a thick-provisioned volume that takes up 15 TB of the gross capacity (created by adding a 7.5 TB volume).

  MDM will allow a total of 300 TB gross to be provisioned, and since 15 TB are already allocated, you can add a thin-provisioned volume of 285 TB gross (by adding a 142.5 TB volume) or a thick-provisioned volume of 15 TB gross.

2. Map volumes. Designate which SDCs can access the given volumes. This gives rise to the following:
   - Access control per volume exposed
   - Shared nothing or shared everything volumes
     Once an SDC is mapped to a volume, it immediately gets access to the volume and exposes it locally to the applications as a standard block device. These block devices appear as `/dev/sciniX` where *X* is a letter, starting from "a."

     For example:

     - `/dev/scinia`
     - `/dev/scinib`
   - When a volume is defined on an AIX SDC, one device is created with the following pathnames:
     - A block device, named `/dev/scinidX...n`, where *X* is a number, starting from "0."
     - A raw device, named `/dev/rscinidX...n`, where *X* is a number, starting from "0."

     In general, mapping SDCs to AIX raw devices will yield best performance. If you are using the device to create a filesystem, use the block device.
   - The maximum amount of partitions for the scini disk is 15.
3. The maximum amount of volumes that can be mapped to an SDC is listed in the "Product capabilities and system" table.

   Note: SDC mapping is similar to LUN mapping, in the sense that it only allows volume access to clients that were explicitly mapped to the volume.

This is the end of the system setup.

Implementing VxFlex OS over a virtual system

This section provides an overview of how VxFlex OS is implemented in a virtualized environment.

VxFlex OS implementation in an ESXi-based system

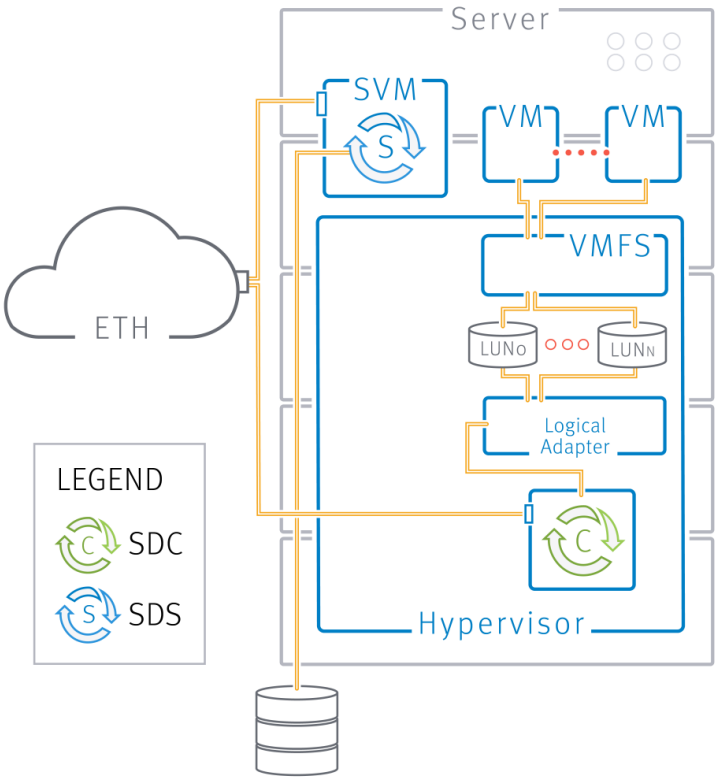
In the VMware environment, the MDM and SDS components are installed on a dedicated SVM, whereas the SDC is installed directly on the ESXi host.

Implementation

Note:
Installing the SDC on the ESXi host requires a restart of the ESXi host.

This implementation is illustrated in the following figure:

Figure 10 VxFlex OS implementation on ESXi



The LUNs in the previous figure can be formatted with VMFS, and then exposed using the ESXi host to the virtual machine, or can be used as RDM devices. When the LUNs are used as RDM devices, the VMFS layer is omitted.

Installation in a VMware environment is performed using the vSphere plug-in.

Device management

Devices can be managed in the following ways:

- VMDirectPath I/O

Device management is performed via the SVM, yielding the best I/O performance. Devices are added to the system after the deployment. On factory-installed VxFlex Ready Node servers, the vSphere plug-in configures DirectPath on the ESXi servers and adds devices to the system. On other servers, you must enable VMDirectPath manually to the ESXi server, then use the plug-in to add devices.

Requirements:

- When DirectPath is configured, all servers in the system must use DirectPath. For VxFlex Ready Node or VxRack Node 100 Series systems, this can be configured with the VxFlex OS vSphere plug-in. For other servers, use the vSphere client to configure each ESXi host manually.
- If the host has multiple controllers, you must configure DirectPath on that host manually, not with the plug-in. After that is done, you can use the plug-in to deploy VxFlex OS on that host and to add devices to it.
- All devices on all servers must not have any other use (not VMDK, RDM, or be part of a datastore).
- PowerEdge 14G servers with any NVME devices cannot be added in a DirectPath-based system. Use RDM-based, instead.
- ESXi boot device requirements:
  - Must be on a separate controller or connected directly to the board.
  - SATADOM and M2 boot devices are supported. These devices allow the creation of a datastore on the system disk, which is needed to host the SVM.
  - USB boot devices are not supported.

Note: DirectPath architecture is the recommended best practice for RAID and SAS Controller managed drives.

- RDM

  Using RDM mapping, a device is created on the SVM that points to the physical disk on the ESXi.

  You can add RDM devices that are connected through a physical RAID controller. If a local RDM is not connected via a RAID controller, it may not be supported. To ensure the compatibility of these devices, you can add them as VMDK, or you can select **Force RDM on device with unknown RDM support**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*. Enable this option before beginning the deployment.

  Before enabling this feature, contact Customer Support.

- VMDK

  A new datastore is created, with a VMDK, and the VMDK is added to the SVM. VxFlex OS requires thick provisioning, so this process can take a long time.

  In almost all cases, RDM is the preferred method to add physical devices. Use the VMDK method only in the following scenarios:

  - The physical device does not support RDM.
  - The device already has a datastore, and the device isn't being completely used. The excess area that is not already being used will be added as a VxFlex OS device.

    Note: To use VMDK, select **Enable VMDK creation**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*.

System size

If you are deploying a very large VxFlex OS system (hundreds of nodes), you can increase the default parallelism limit (default: 100), thus speeding up the deployment. This is dependent on the processing power of the vCenter.

To increase the parallelism limit, use the plug-in **Advanced settings**, as described in the "Advanced settings options" section of the *VxFlex OS Deployment Guide*.

Pre-deployment considerations

You should take these considerations into account before deploying the system:

- Do you want to use separate networks for data and management (recommended), and which IP addresses will you use for the SVMs and VMkernels?
- Are there flash devices that will be added to the SDS?
- Do you want to create Fault Sets? See the requirements in "Fault Sets".

Post-deployment considerations

You should take these considerations into account after deploying the system:

- After deployment is complete, set all SVMs to start automatically with the system. Do not set SVMs under the VMware resource-pool feature.
- In a DirectPath environment, after deploying the system, you must add devices to the SDS.

Other vSphere plug-in features

The plug-in's deployment wizard can be used to register existing VxFlex OS systems in vSphere, to add SDS server nodes, or to extend the existing system.

The plug-in can be used to provision VxFlex OS nodes.

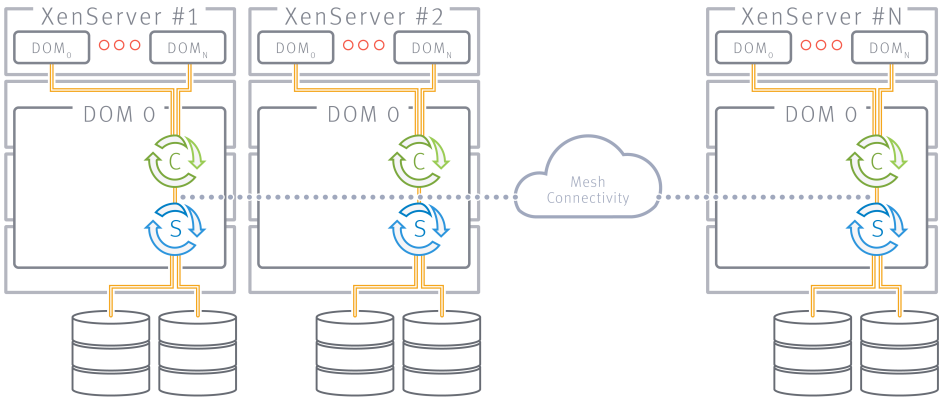
Note: When using DirectPath, also known as PCI-passthrough on the SVM, the following alert appears in vCenter: "Virtual Machine memory Usage". This is a VMWare limitation and can't be resolved. Refer to the VMWare KB for more details:https://kb.vmware.com/s/article/2149787

Xen implementation

This section demonstrates the architecture of Xen virtual machine.

In a Xen environment, both the SDC and SDS are installed in Dom0 as would be on a physical node. Dom0 accesses the storage media through the SDS and exposes volumes based on VxFlex OS through the SDC.

Figure 11 VxFlex OS Xen virtual machine architecture



Information on provisioning in a Xen environment is described in this guide.

Maintenance

Maintenance of VxFlex OS is primarily limited to configuration changes of the physical and virtual layers. It requires minimal user attention. When maintenance or planned restart of an SDS is required, the maintenance mode feature can be used to streamline system operation.

Maintain the physical layer

Adding/removing hardware units and configuring them into VxFlex OS can result from scale out, hardware failure, OS patching/upgrade and hardware expansion.

In the physical layer, maintenance is limited to adding and removing hardware units and configuring them into VxFlex OS. These operations are usually a result of:

- Scaling out when there is a need for additional capacity. This usually results in adding more storage media to the existing servers, or adding additional servers.
- Hardware failure. In cases where there is a hardware (storage media or server) failure and it needs to be replaced.

  In all of the above cases, the operation will require adding or removing storage capacity from the system. In some cases, it may include adding or removing an entire server, and its associated storage media, from the configuration. As far as VxFlex OS is concerned, all of these activities translate to SDS reconfigurations.

  If the removed server is an SDC node, or the server to be added requires exposing storage locally, SDC reconfiguration will happen as well.
- Adding or removing storage media. Add or remove the media from the SDS with which it is associated. VxFlex OS will redistribute the data accordingly and seamlessly.
- Adding or removing a node. Add or remove the SDC and SDS residing on the node. VxFlex OS will redistribute the data accordingly and seamlessly.
- OS patching \ Upgrade or Hardware expansions. Graceful shutdown\reboot operations with return to fully operation state flow.
- Instant maintenance mode enables you to restart a server that hosts an SDS, with minimal impact on VxFlex OS, thus bypassing the disruption and effort caused by disorderly shutdown, Protection Domain shutdown, and orderly shutdown.

Instant maintenance mode

Using instant maintenance mode, you can restart a server hosting an SDS without requiring shutdown or interrupting application I/Os. Storing all writes created during maintenance to a dedicated Fault Set prevents data loss in case of a single failure.

Instant maintenance mode enables you to restart a server that hosts an SDS, with minimal impact on VxFlex OS, thus bypassing the disruption and effort caused by disorderly shutdown, Protection Domain shutdown, and orderly shutdown.

Whereas VxFlex OS always uses two copies of user data, invoking maintenance mode introduces an additional copy that stores all writes created during maintenance to an SDS or Fault Set (created during maintenance) in both a primary location and a new location. This copy prevents data loss if a single failure occurs.

When the SDS or Fault Set is returned from maintenance mode, only the new writes are required to be resynchronized, thus minimizing data transfer during and after the update.

Instant maintenance mode does not interrupt application I/Os; it can be run on any amount of members of a Fault Set; and it can run in parallel on different Protection Domains. While an SDS is

in maintenance mode, most VxFlex OS operations (like adding a volume) cannot be performed in the Fault Set, Protection Domain, or Storage Pool in which the SDS and its devices reside.

To invoke maintenance mode, the following conditions are required:

- Only one Fault Unit (or standalone SDS) can be in maintenance mode at any given time.
- No other SDSs can be in degraded or failed state (force override can be used).
- There must be adequate space on other SDSs for the additional backup (force override can be used).

  Note: Use of force override options when entering maintenance mode can lead to data unavailability while maintenance mode is activated.

  While an SDS is in maintenance mode, it can be shut down with no danger to data.

Maintaining the virtualization layer

This section lists the operations that can be performed on volumes exposed by the VxFlex OS virtual SAN.

The following operations may be performed on volumes that are exposed by the VxFlex OS virtual SAN:

- Add or remove a volume:

  Create or delete a volume in the system.
- Increase volume size:

  Add capacity to a given volume, as needed. The change in volume size occurs seamlessly without interrupting I/O.
- Map and unmap volumes to an SDC:

  This enables or disables access to a volume by an SDC, and thus by an application residing on the same node.
- MDM operations:
  - Move the MDM role to another role
  - Add MDM VIP
  - Upgrade of MDM, SDS, SDC and LIA

Management tools

This section lists the available management clients that can be used with VxFlex OS. Many of VxFlex OS actions can be performed using more than one tool.

You can provision, maintain, and monitor VxFlex OS using the following management clients:

- Command Line Interface (CLI)

  The CLI enables you to perform the entire set of configure, maintain, and monitor activities in VxFlex OS.
- Graphical User Interface (GUI)

  The VxFlex OS GUI enables you to perform standard configure and maintain activities, as well as to monitor the storage system's health and performance. You can use the VxFlex OS GUI to view the entire system, and then drill down to different elements.
- VMware plug-in (plug-in)

  The plug-in enables you to perform basic provision and maintain activities in the VMware environment. In addition, the plug-in provides a wizard to deploy VxFlex OS in the VMware environment.

- OpenStack

  VxFlex OS provides a Cinder driver, which enables interoperation between VxFlex OS and an OpenStack cloud operating system.
- REST gateway

  A REST API can be used to expose monitoring and provisioning via the REST interface. The REST server is installed as part of the VxFlex OS Gateway

Many VxFlex OS activities can be performed in more than one management tool.

The following tool is also provided:

- VxFlex OS Installer.The VxFlex OS Installer is used for installing VxFlex OS, upgrading, and uninstalling components, as well as running the get-info operation. The VxFlex OS Installer is installed as part of the VxFlex OS Gateway.

VxFlex OS VASA architecture

VxFlex OS has implemented VMware's vSphere API for Storage Awareness (VASA). VxFlex OS's VASA enables control of VxFlex OS storage to be handled by the vCenter administrator. VxFlex OS's VASA uses VMware vSphere Virtual Volumes (vVols) to enable VMs hosted on ESXis in the vCenter to have their storage mapped directly to storage in VxFlex OS. VxFlex OS's implementation uses VASA 2.0 and vVols 1.0. There can be one or three instances of VASA.

The following are the different components of VxFlex OS's VASA:

- Virtual Volumes (vVols): A VMware object type that allows the vSphere Admin to provision VMs without depending on the Storage Admin.
- vVol datastores: A type of VMware datastore, in addition to VMFS and NFS datastores, which allow vVols to map directly to the storage system. vVol datastores are more granular than VMFS and NFS datastores, enabling VMs or virtual disks to be managed independently. You can create vVol datastores based on one or more underlying storage pools and then allocate the pool to be used for the vVol datastore and its associated vVols.
- Storage containers: A logical abstraction on to which vVols are mapped and stored. In VxFlex OS's VASA, the storage container is mapped to a VxFlex OS Storage Pool. vSphere maps storage containers to VVol datastores and provides applicable datastore level functionality.
- Storage policies: Storage policies are used to ensure that VMs are placed on storage that guarantees a specific level of performance.
- VASA Storage Virtual Machine (SVM): The virtual machine that runs the VASA.
- VASA-database: A database created to store the VVols metadata.

VASA allows for automatic mapping of storage. When a new VM is created, a request for storage passes through the VASA. The VASA sends the request to the MDM to create the volume. The MDM creates the volume and passes the volume ID back through the VASA to the ESXi server. Mapping volume requests also pass through the VASA as the MDM maps the volume the VM runs on to the ESXi. For every VM that is created, a minimum of two vVols are created. One is for the VM metadata and one is for the VM disk. When the VM is powered on, another vVol, called the swap vVol, is created. The swap vVol is removed when the VM is powered off. The VASA is used only for storage management; no I/Os are sent through the VASA.

VxFlex OS VASA's limitations and prerequisites

Note the following limitations when using VxFlex OS's VASA.

Limitations

- Deployment of VxFlex OS's VASA is not prevented on ESXi 5.5, despite the fact that VMware no longer supports ESXi 5.5.

- The maximum number of volumes is 200K (4,096 mapped volumes per ESXi).
- The VASA can be mapped to a single instance of VxFlex OS.
- A cluster may contain a single VASA provider, three Vasa providers, or none. Any other configuration is illegal.
- The VASA provider may be installed only on a node which contains an SDC.
- The VASA should be registered with a single vCenter.
- The VASA provider hostname must begin with a letter character.
- The maximum snapshot hierarchy of a vVol is 31.

Before you begin

Ensure that:

- Oracle Java Development Kit 1.8.x must be installed on every host that will run the VASA provider.
- For each VASA VM, the DNS server is configured and all ESXis, vCenters, and VASA Storage Virtual Machines (SVM) are registered on it.
- Each VASA VM's hostname must match the VASA FQDN.
- All components of your environment, including ESXis and vCenters, have their time synchronized.
- A Storage Data Client (SDC) is installed on the ESXi on which you want to deploy VASA.
- VxFlex OS is deployed and registered in the vCenter.

Configuring direct attached storage (DAS)

When adding DAS to SDA, it is recommended to configure the raw devices as stand-alone devices. Upon adding a device to an SDS VxFlex OS verifies that the device is clear and returns an error if it is otherwise.

VxFlex OS works with any free capacity—internal or direct-attached devices, either magnetic hard disk drives (HDD) or flash-based devices such as solid state drive (SSD) and PCIe cards. Although VxFlex OS can work with any device topology, it is recommended to configure the raw devices as stand-alone devices.

Device data is erased when devices are added to SDS. When adding a device to an SDS, VxFlex OS will check that the device is clear before adding it. An error will be returned, per device, if it is found not to be clear. You can override this check by using the force device takeover option.

The following devices are considered to be not "clear," and thus cannot be added to SDS:

- Linux - A complete device with either a filesystem or partition, or a partitioned device with a filesystem.
- ESXi - Same as above, depending on the OS of the SVM where the SDS is installed.

Limitations:

- SAN devices will not be prevented from being added.
- Devices in an LVM group cannot be added to an SDS.
- Within the database devices, only Oracle ASM devices can be detected and blocked.

Note:
If the server has a RAID controller, VxFlex OS prefers to use the controller's caching abilities for better performance, but is better utilized when all devices are configured as stand-alone

(i.e. setting each of the devices to RAID-0 separately). For HDD devices, it is recommended to enable RAID-controller caching. As for flash devices, it depends on the device behavior.

Note:
For HDDs: It is recommended to use RAID-controller caching when available as follows:

- READ/WRITE: if cache is battery-backed
- READ ONLY: if cache is NOT battery-backed

For flash devices (e.g. SSD): Depends on the device

Product capabilities and system limits

The following table summarizes VxFlex OS capabilities.

Table 5 Product limits

Item VxFlex OS	Limit
System raw capacity	Up to 16 PB . Derived from minimum device size
Device size	Min: 90 GB (MG), 128 GB (FG) Max: 8 TB
Minimum Storage Pool capacity	300 GB
Volume size	Min: 8 GB Max: 1 PB
Maximum number of volumes/snapshots in system	262,143[a]
Maximum number of volumes/snapshots in Protection Domain	32,768
Maximum number of volumes + snapshots in single VTree	128
Maximum capacity per Protection Domain	8 PB
Maximum capacity per SDS	128 TB
Max SDSs per system	512A
SDSs per Protection Domain	128[a]
Maximum devices (drives) per SDS server	64[b]
Maximum disks per Protection Domain	8192 (128x64)
Maximum devices (drives) per Storage Pool	300 [a] for VxFlex OS 3.0.0.1 and up 240 for VxFlex OS 3.0
Minimum devices (drives) per Storage Pool	3, on different Fault Sets
Maximum SDCs per system	1024
Maximum volumes that can be mapped to a single SDC	8192
Maximum Protection Domains per system	256

Table 5 Product limits (continued)

Item VxFlex OS	Limit
Maximum Storage Pools	1024
Maximum Storage Pools per Protection Domain	64
Maximum Fault Sets per Protection Domain	64 [a]
Maximum IP addresses per server (MDM and SDS)	8
RAM Cache	128 MB—300 GB
Fine-Granularity max compression	10X of raw capacity
Maximum Snapshot Policies per V-tree	60
Maximum Snapshot Policies	1000
Maximum user accounts	256 [a]
Concurrent GUI/REST/CLI clients logged-in	128
Maximum number of syslog clients	16

a. If more are needed, contact Customer Support.
b. On VMware servers, the maximum devices per SDS is 59.

PART 3

System Requirements

This section includes an overview of the system requirements.

Chapter 3, "System Requirements"

CHAPTER 3

System Requirements

The following topics describe the system requirements for the VxFlex OS system.

- System requirements 66
- License for VxFlex OS 72
- Activate the license 72
- Activating entitlements and installing a license file 74
- License file example 78
- Licensing error messages 78

System requirements

This section lists the requirements for system components.

This section is specific to VxFlex OS software deployments.

For VxFlex Ready Node or VxRack Node 100 Series systems, refer to your product's Hardware Configuration and Operating System Installation Guide.

VxFlex OS cluster components

Each VxFlex OS cluster includes the following components. Before deploying VxFlex OS, ensure that the VxFlex OS environment meets the server requirements for the cluster type.

Three-node cluster

- One Master MDM
- One Slave MDM
- One Tie Breaker
- Minimum of three SDSs (on the same servers as the above components, or on three different servers)
- SDCs, up to the maximum allowed (on the same servers as the above components, or on different servers)

Five-node cluster

- One Master MDM
- Two Slave MDMs
- Two Tie Breakers
- Minimum of three SDSs (on the same servers as the above components, or on three different servers)
- SDCs, up to the maximum allowed (on the same servers as the above components, or on different servers)

For additional information, see the Getting to Know VxFlex OS Guide.

Additional requirements

VxFlex OS Gateway server on a separate server, or together with an MDM or SDS. Do not install the VxFlex OS Gateway on an SDC server.

Physical server requirements

The following table summarizes the requirements of the physical server.

Table 6 Server physical requirements

Component	Requirement
Processor	One of the following: • Intel or AMD x86 64-bit (recommended) • Intel or AMD x86 32-bit (for Xen only)

Table 6 Server physical requirements (continued)

Component	Requirement
	Note: AMD processors are not supported on VxFlex Ready Node and don't support NVDIMM's. For this reason, there is no support for FG storage pools on AMD based servers.

Physical memory	VxFlex OS component requirements:
	• 1.1GB RAM for the Meta Data Manager (MDM)
	• 500 MB RAM for each VxFlex OS Data Server (SDS). This is higher if using Fine Granularity pools.
	  ▪ The size may grow to 19 GB if the maximum capacity of 128 TB is used.
	  ▪ The SDS uses about 760 MB to start and 140 MB when connected to MDM. Then, about 30 MB per TB for connected devices and up to about 10 MB per TB of volumes in the Storage Pool.
	  These numbers represent memory that the SDS process needs to operate per added device / volume. You need to have a bit more memory for the process itself.
	  Note: Various configurations can effect memory capacity, such as release, debug build, logs buffer size, performance profile setting
	• 50 MB RAM for each VxFlex OS Data Client (SDC)
	• 350 MB for LIA

Disk space	• For Linux or XenServer—1 GB for each physical node or Xen hypervisor • For ESXi—10 GB for VMware topologies

Connectivity	One of the following:
	• 1/10/25/40/50/100 GbE network
	• IP-over-InfiniBand network
	Dual-port network interface cards (recommended)
	Ensure the following:
	• There is network connectivity between all components.
	• Network bandwidth and latency between all nodes is acceptable, according to application demands.
	• Ethernet or InfiniBand switch supports the bandwidth between network nodes.
	• MTU settings are consistent across all servers and switches.
	• The following TCP ports are not used by any other application, and are open in the local firewall of the server:
	  ▪ MDM: 6611 and 9011
	  ▪ SDS: 7072 (for multiple SDS, ports 7073-7076)
	  ▪ Light Installation Agent (LIA): 9099
	  ▪ SDBG ports: MDM 25620, SDS 25640, Multiple SDS 25641-25644 (not 25640). These ports are used by VxFlex OS internal debugging tools to extract live information from the system for debugging purposes.
	• The following UDP port is open in the local firewall of the server:

Table 6 Server physical requirements (continued)

Component	Requirement
	▪ SNMP traps: 162 **Note:** You can change the default ports. For more information, see "Changing default ports" in the user documentation.

Supported operating systems

The following is a list of operating systems supported by this version of VxFlex OS.

For the most updated list, see the EMC Simple Support Matrix (ESSM) at https://elabnavigator.emc.com/eln/elnhome.

Table 7 Supported operating systems -VxFlex OS components

Operating system	Requirement
Linux	Supported versions (for exact versions, see the ESSM): • Red Hat Enterprise Linux (RHEL) 6.9, 6.10, 7.5, 7.6 and 8.0 • CentOS - 6.9, 6.10, 7.5, 7.6 and 8.0 • Oracle Linux - 6.9, 6.10, 7.5 and 7.6 • SUSE Linux Enterprise Server (SLES) - 12.4 and 15 • Ubuntu 16.04.6 LTS, 18.0.4.2 LTS • Citrix Hypervisor (XenServer) 7.1.2 LTSR, 7.6 and 8.0 • CoreOS - SDC only (RFcache is not supported) **Note:** Before deploying SDC or RFcache on Ubuntu servers, you must prepare the environment, as described in the *Deploy VxFlex OS Guide*. Packages required for all components, all Linux flavors: • numactl • libaio Additional packages required for MDM components: • bash-completion (for SCLI completion) • Latest version of Python 2.X When installing the MDM component on RHEL 6 hosts, set the shared memory parameter in the /etc/sysctl.conf file to at least the following value: kernel.shmmax=209715200. To use this value, type the `sysctl -p` command. To use the secure authentication mode, ensure that OpenSSL 64-bit v1.0.1 or later (v1.1, however, is not supported) is installed on all servers in the system. • `libopenssl1_0_0-1.0.1g-0.40.1.x86_64.rpm` • `openssl1-1.0.1g-0.40.1.x86_64.rpm` To use LDAP, ensure that OpenLDAP 2.4 is installed on all servers.
Windows	Supported versions:

Table 7 Supported operating systems -VxFlex OS components (continued)

Operating system	Requirement
	• Only SDC, LIA, Gateway and AMS are supported components on Windows • Windows Server Core Editions (2012 R2, 2016, or 2019) are not supported. (For VxFlex Ready Node, 2012 is not supported.) • For VxRack Node 100 Series, only 2012 R2 is supported. • To install SDC, ensure that Microsoft Security Update KB3033929 is installed. To use the secure authentication mode, ensure that these are installed on all servers in the system: • OpenSSL 64-bit v1.0.1 or later (v1.1, however, is not supported) • Visual C++ redistributable 2010 package, 64-bit The Microsoft Windows "Storage Spaces" feature (available from Windows 8.1 and later), is not supported in Microsoft Failover cluster running on VxFlex OS volumes.
Hypervisors	VMware ESXi operating systems: 6.5 U3 or 6.7 U2, managed by vCenter 6.5 or 6.7.
AIX	For SDC only - 7.2
VMware	6.5 and 6.7
MacOS	For GUI only - 10.12, 10.13

SVM memory requirements

SVM memory requirements and formula for memory allocation rules assist you in determining the amount of memory needed for optimal system performance.

The memory allocation formula:

Component	Memory allocation rules
Base SVM	• 350 MB
MDM (Master/Slave)	• 470 MB + (500 KB * 8 TB of volume capacity) + (1.44 KB * number of volumes) + (4 KB * number of SDS devices) • Maximum supported volumes: 256 K **Note:** 1.1 GB usually covers most use cases.
Tie Breaker MDM	• 50 MB
SDS	For Medium Granularity storage: • 900 MB + 30 MB * <number of TB of connected devices > + 10 MB * <number of TB of connected volumes> NOTE, various configurations can effect, memory capacity. • For SDS high performance profile, add 195 MB. These numbers represent memory that the SDS process needs to operate per added device / volume. It needs to have additional memory for the process itself. Note that various configurations can also affect memory capacity. Processes such as rebalancing, rebuilds, and snapshots also require memory capacity.

Component	Memory allocation rules		
SDC	• 132 KB + 23 MB * (number of MDMs) + 25 KB * (number of SDSs) + 1.5 KB * (number of volumes) + 16 B * (number of volume blocks) + 24 KB * (8 TB of volume capacity) • Volume blocks: 1 GB storage = 8 volume blocks		
LIA	• 350 MB		
RFcache	• 16 * (cache_size/page_size) • Commonly-used sizes:		
	RFcache page size	RFcache memory requirement, if the cache device is 800 GB	RFcache memory requirement, if the cache device is 1.6 TB
	64 K	200 MB	400 MB
	32 K	400 MB	800 MB
	16K	800 MB	1.6 GB
	8 K	1.6 GB	3.2 GB
	4 K	3.2 GB	6.4 GB

External SDC support

In addition to being supported on all VxFlex OS operating systems, SDC can be deployed on external servers.

Component	Requirement
Supported external servers	• UNIX: AIX 7.2 • Linux: 4.x/5.x

VxFlex OS GUI server requirements

Refer to VxFlex OS Support Matrix for VxFlex OS GUI server requirements:

Component	Requirement
Supported operating systems	• Windows: ▪ 7, 2008 R2, 10, 2012 or 2012 R2, 2016. Server Core editions are not supported. • Linux: ▪ CentOS 6.x-7.x, Oracle Linux 6.5/7.x ▪ Red Hat 6.x-7.x ▪ SUSE 11.3, 11.4, 12, 12.1, 12.2 ▪ Ubuntu 14.04, Ubuntu 16.04

Component	Requirement
Other	• For SUSE 12, see "Installing Java on SUSE 12 servers." in the *VxFlex OS Deployment Guide* or the *VxFlex OS User Guide*.) • Screen resolution: 1366 x 768 minimum

VxFlex OS Gateway server requirements

Refer to VxFlex OS Support Matrix for VxFlex OS Gateway server requirements:

Component	Requirement

Supported operating systems	• Windows 2019, 2012 R2, or 2016, including the Visual C++ redistributable 2010 package, 64-bit. Server Core editions are not supported.
	• Linux:
	  ▪ CentOS 6.x-7.x
	  ▪ Oracle Linux 6.5/7.x
	  ▪ Red Hat 6.x-7.x
	  ▪ SUSE 12.x
	  ▪ Ubuntu 14.04, Ubuntu 16.04, Ubuntu 18.04
	Every server requires 2 cores and a minimum of 3 GB available RAM.

Connectivity	The following TCP ports are not used by any other application, and are open in the local firewall of the server: 80 and 443 (or 8080 and 8443). You can change the default ports. For more information, see "Changing default ports" in the user documentation.
Supported web browsers	• Internet Explorer 11, or later • Firefox, version 65.0.1, or later • Chrome, version 72, or later
Other	• For a VxFlex OS Gateway server on a Windows node, the Windows Management Instrumentation service must be enabled on the VxFlex OS Installer server and on all Windows VxFlex OS nodes. • Do not install the VxFlex OS Gateway on a server on which RFcache will be enabled or on which SDC will be installed. • The VxFlex OS Gateway server must have connectivity to all the nodes that are being installed. If you are using separate networks for management and data, the server must be able to communicate with both networks.

Other requirements

This section lists additional requirements for VxFlex OS.

VxFlex OS requires that you use a minimum of three SDS servers, with a combined free capacity of at least 300 GB. These minimum values are true per system and per Storage Pool.

License for VxFlex OS

This topic describes activating the system after registration with the license obtained either from the LAC email or from the Dell EMC support site.

After installing VxFlex OS, to activate the system, you must register the system with a license that can be obtained from the LAC email you received after your purchase or you can access the license through the Dell EMC support site by entering your Sales Order number.

After the license has been applied, you can continue to configure and manage your VxFlex OSenvironment.

The license is installed on the MDM cluster.

Activate the license

Activate the license in order to use your system in a production environment.

About this task

VxFlex OS installations are enabled to be fully functional, for non-production environments.

Using VxFlex OS in a production environment requires a license. The license is installed on the MDM cluster, using the SCLI command `--set_license`.

To obtain a license for production use, and to receive technical support, open a service ticket at https://support.emc.com.

VxFlex OS licenses are purchased by physical device capacity (in TB). You can activate your licensed capacity over multiple VxFlex OS systems—each system with its unique installation ID.

You download VxFlex OS licenses from the Dell EMC Software Licensing Central website, using the procedures described in "Activate entitlements and download a license file". Then, you install the licenses on VxFlex OS, as described in “Install the license”.

You can view current license information using the CLI or the VxFlex OS GUI.

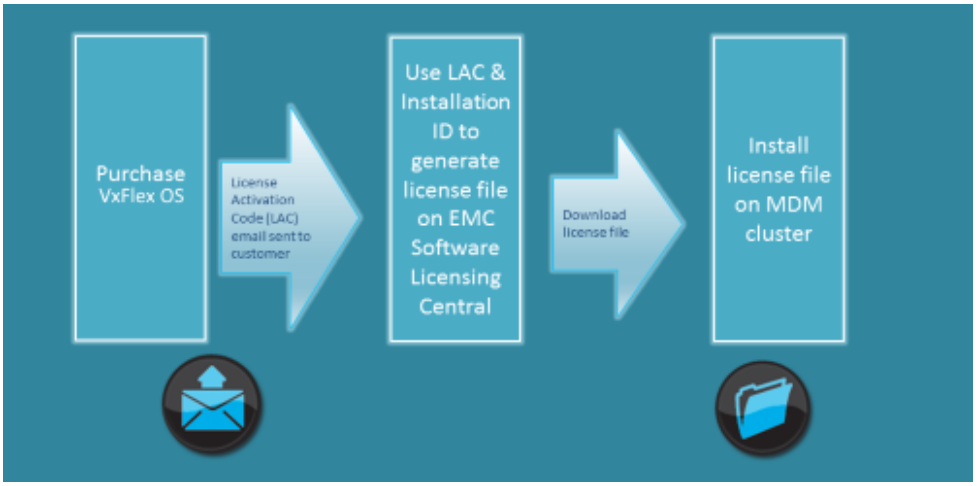
The following steps summarize the licensing process.

Procedure

1. Purchase VxFlex OS, and receive a License Authorization Code (LAC) email with a link to the licensing site.

   If you do not have the LAC email, you can search for the LAC number from the Dell EMC Software Licensing Central website, by entering the Sales Order number and using the **Search Entitlements** option.

2. Retrieve the installation ID from VxFlex OS.
3. Click the link in the LAC email, and use the online wizard to complete the entitlement activation process.
4. Save the license file, and install it using the CLI.



The following table describes VxFlex OS eLicensing terminology.

Table 8 eLicensing terminology(continued)

Term	Description
Dell EMC Online Support	The online support portal, http://support.emc.com, contains product support information and links to the Software Licensing Central web site.
Entitlements	The Dell EMC Software Licensing Central web site lists the entitlements (usage rights) that you have purchased, that you can activate for a specific host machine.
LAC email	Email sent to a customer who has purchased an Dell EMC product, containing a License Authorization Code (LAC), which is needed to complete the entitlement activation process on the Dell EMC Software Licensing Central web site.

When you purchase a license entitlement for VxFlex OS, a License Authorization Code (LAC) email is sent to you, or to your purchasing department. If you cannot find the LAC email, you can use the Software Licensing Central website to find your license entitlements.

The following figure shows a sample LAC email:

Figure 12 Licensing LAC email

Dear EMC Software User,
Thank you for choosing EMC software. Your EMC Software License Authorization Code (LAC) is [redacted].
You must redeem this LAC for license keys to activate your software. Please protect your LAC like you would any other license key to prevent anyone from improperly activating your software.

Activating Your Software
1. Click here or copy and paste the following URL (https:[redacted]
LAC=[redacted]) into a web browser to activate your entitlements.
2. You will be prompted to log in. (New users should follow the new member registration steps).
3. Follow the on-screen instructions.

Downloading Your Software
1. Click here or copy and paste the following URL (https://ngtest-ci.emc.com/downloads/) into a web browser to download your software.
2. You will be prompted to log into EMC's Online Download Service Center (New users should follow the new member registration steps).
3. Enter the product name in the search field to find the software you wish to download.

License Authorization Code: RR91DYQLF9RTXZT9RMQH

Product #	Title	Quantity
456-106-154	EMC SCALEIO SOFTWARE CAPACITY=CB	250
456-106-155	EMC SCALEIO ENTERPRISE FEATURES=CB	250

If you have any questions about your sales order please contact your EMC Account Representative or your Authorized Reseller.

Activating entitlements and installing a license file

VxFlex OS licenses are assigned to VxFlex OS systems, each of which is identified by a unique installation ID. Use the VxFlex OS installation ID along with your LAC to activate the entitlement and then download the license file. Then install this file in your MDM cluster.

Activate an entitlement and download the license file

This section describes how to activate the purchased entitlement, and download the license file.

About this task

VxFlex OS is procured by total capacity, but you can activate portions of this total capacity over multiple VxFlex OS installations. For example, your purchase order may have been for 1000 TB. Your License Authorization Code (LAC) will entitle you to activate all, or part of that. You can activate 500 TB for one VxFlex OS installation, and leave the rest for another activation, for the same, or for a different system.

To activate the entitlement, perform the following:

Procedure

1. Obtain the VxFlex OS installation ID:
   - Using the CLI:
     Log in to the CLI, and then run the following command:

     ```
     scli --query_license
     ```

     The installation ID is displayed:

     ```
     Installation ID: 0123456789abcdef
     ```

     Note:
     Actual command syntax is operating-system dependent. For more information about logging in and running commands, see the *VxFlex OS CLI Reference Guide*.
   - Using the VxFlex OS GUI:

From the top right of the main window, open the drop-down menu that appears next to the user name, and select **About**.

The installation ID is displayed in the **About** window.

2. If you have the LAC email, skip to step 4.
3. If you do not have your LAC email, perform the following:
   a. From the Dell EMC support website, browse to the Software Licensing Central system:
      a. Open the support website: http://support.emc.com.

         If you are a new user, create a new user account.
      b. From the **Support Tasks** list, click **Manage Licenses and Usage Intelligence**.
      c. From the software list, click **VxFlex OS** . The **Powerlink Licensing** website is displayed.
      d. Click **View Entitlements**. The **Search Entitlements** screen appears.

   b. Type the Sales Order number, then click **Search Entitlements**.

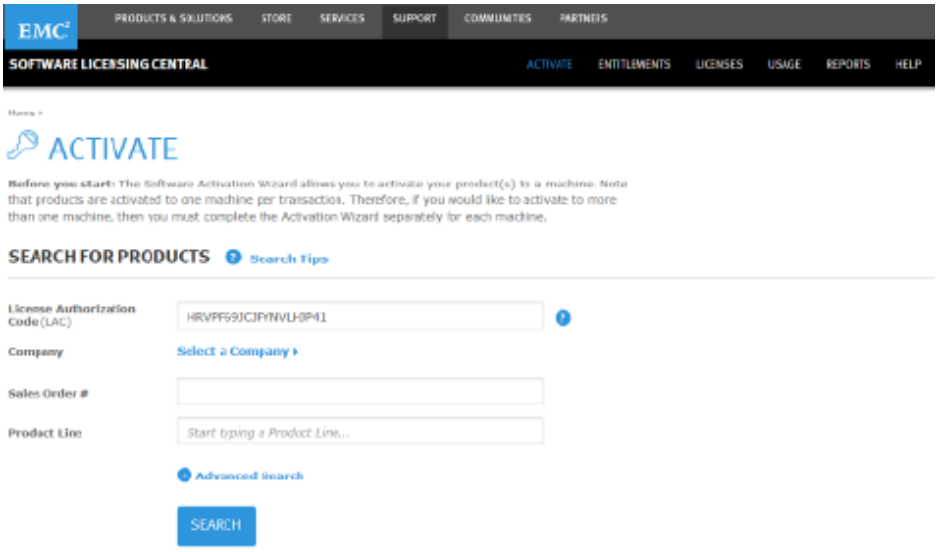
      A list of entitlements is displayed.

   c. Locate the entitlement to activate, and choose **Options** > **Activate**.

   The **Powerlink Licensing—Search Entitlements to Activate** screen appears. Skip to step 6.

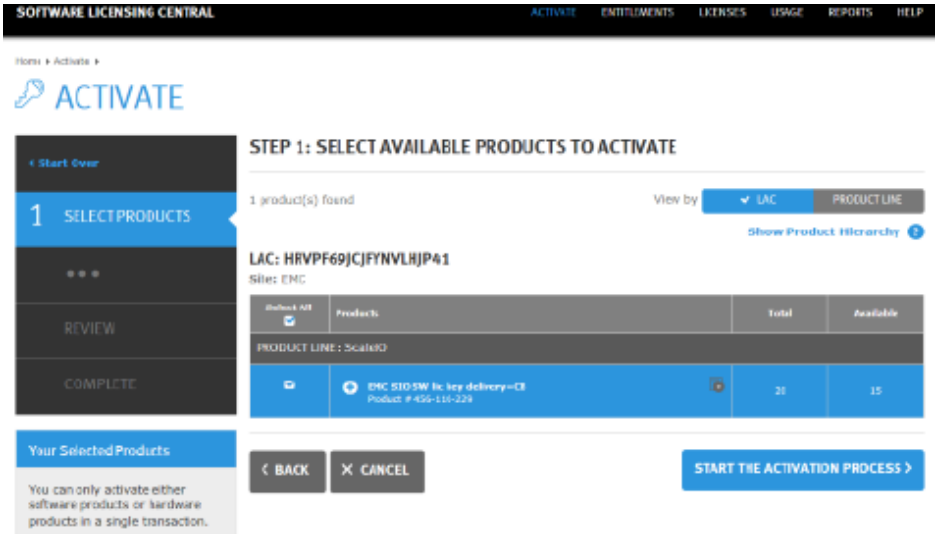
4. If you have your LAC email, perform the following:
   a. Click the link in the LAC email, and log in.

      The **Activate—Search for Products** screen appears:



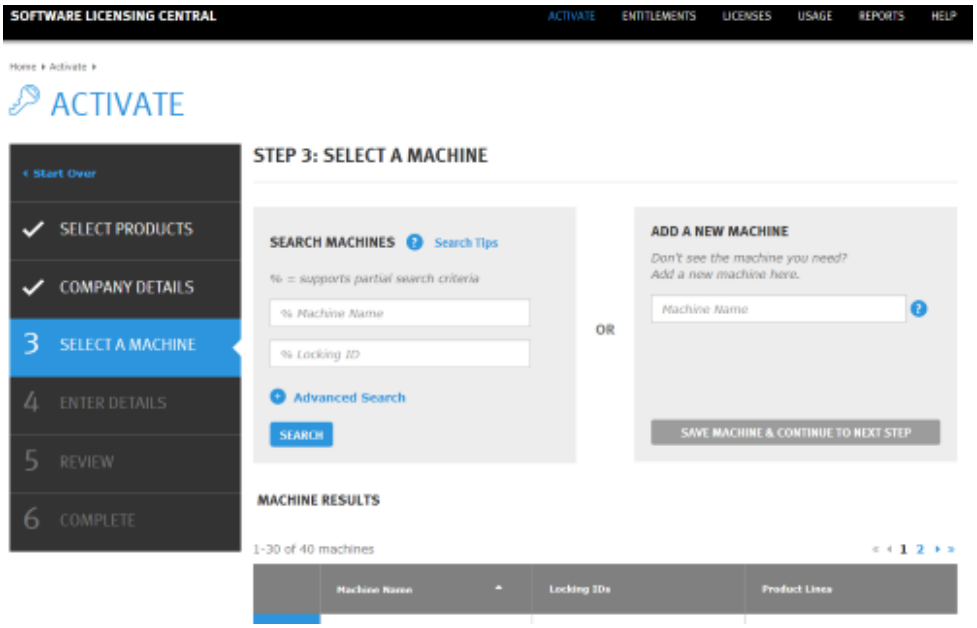
   b. Enter your LAC code, or search by Sales Order number, then click **Search**.

      The **Select Products** screen appears:



5. In the **Select Products** screen, select the product to activate, and click **Start the Activation Process**.
6. In the **Company Details** screen, confirm (or update) company information, and click **Select a Machine**.

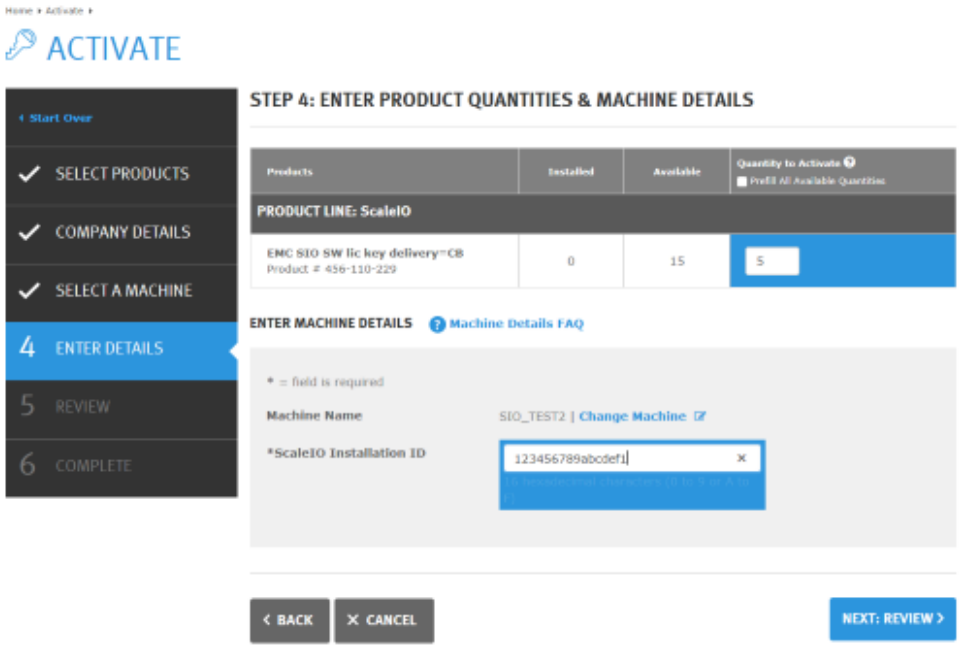
   The **Select a Machine** screen appears:



7. In the **Select a Machine** screen, select a machine on which to activate the product in one of these ways:
   - Click **Search** to locate an existing machine (one on which a Dell EMC product was previously activated).
   - Add a new machine name, then click **Save Machine & Continue**.

   In the context of the activating process, a machine is a VxFlex OS system, which could comprise multiple servers.

   The **Enter Details** screen appears:



8. In the **Enter Details** screen, enter the following:
   - **Quantity** (in TB) to activate on this machine.

     To allocate the available capacity over multiple machines, select less than the full amount available, and repeat the activation process on the other machines.
   - VxFlex OS Installation ID, from the beginning of this procedure.
9. Click **Next**.
10. In the **Review** screen, you can review your selections.

    The license key will be emailed to the username that is logged in to the licensing system. To send it to more recipients, click **Email to more people** and enter their email addresses.
11. Click **Activate**.

Install the license

Use SCLI to install the license.

About this task

Actual command syntax is operating system-dependent. For more information, see the *VxFlex OS CLI Reference Guide*.

Procedure

1. Using SCLI, run the command:

   ```
   scli --set_license --license_file <license_file>
   ```

   where *<license_file>* is the full path to the license file.

   ```
   scli --set_license --license_file /tmp/0239SH4SS89023T6.lic
   ```

Results

The VxFlex OS license is now installed on the MDM cluster.

You can view license information using the `query_license` command, and from the **About** menu in the VxFlex OS GUI.

License file example

The following figure illustrates a license file with a license for 200TB of capacity.

Figure 13 License file example

```
Header
        ##############################################################
        # EMC License File
        # Activation Date:  Apr 10, 2014 08:49:41 AM
        # Activated By:  robert grosso
        # Type:UNSERVED
        ##############################################################
Base capacity
        INCREMENT SIO_BASE EMCLM 1.0 permanent uncounted \
                VENDOR_STRING=CAPACITY=200;CAPACITY_UNIT=TB;installation_id=;SWID=SL6LC2B6HB4YSK;PLC=SIO; \
                HOSTID=ANY dist_info="ACTIVATED TO SIOTEST CO." ISSUER=EMC \
                ISSUED=10-Apr-2014 NOTICE="ACTIVATED TO License Site Number: \
                PTA10APR20141109559" SN=2169155 SIGN="004A FD6C 87EC 2F63 248F \
                FE9B A852 C700 86DB 8332 21F2 9C72 5744 759C D6FE"
Enterprise features
        INCREMENT SIO_SNAPSHOTS EMCLM 1.0 permanent uncounted \
                VENDOR_STRING=CAPACITY=200;CAPACITY_UNIT=TB;installation_id=;SWID=SL6LC2B6HB4YSK;PLC=SIO; \
                HOSTID=ANY dist_info="ACTIVATED TO SIOTEST CO." ISSUER=EMC \
                ISSUED=10-Apr-2014 NOTICE="ACTIVATED TO License Site Number: \
                PTA10APR20141109559" SN=2169155 SIGN="004C 6427 45B9 3BC4 2656 \
                EC20 3A51 6300 6466 F7A7 566A 59AD 0B8D 2BEA DB10"
        INCREMENT SIO_QoS EMCLM 1.0 permanent uncounted \
                VENDOR_STRING=CAPACITY=200;CAPACITY_UNIT=TB;installation_id=;SWID=SL6LC2B6HB4YSK;PLC=SIO; \
                HOSTID=ANY dist_info="ACTIVATED TO SIOTEST CO." ISSUER=EMC \
                ISSUED=10-Apr-2014 NOTICE="ACTIVATED TO License Site Number: \
                PTA10APR20141109559" SN=2169155 SIGN="0069 FC1E 0AF5 FEFF F20C \
                EBB2 146A 7A00 3C7E 42B0 6D0F 7D78 4560 9B3C E534"
        INCREMENT SIO_OBFUSCATION EMCLM 1.0 permanent uncounted \
                VENDOR_STRING=CAPACITY=200;CAPACITY_UNIT=TB;installation_id=;SWID=SL6LC2B6HB4YSK;PLC=SIO; \
                HOSTID=ANY dist_info="ACTIVATED TO SIOTEST CO." ISSUER=EMC \
                ISSUED=10-Apr-2014 NOTICE="ACTIVATED TO License Site Number: \
                PTA10APR20141109559" SN=2169155 SIGN="00D7 202C 9FCA E6F0 F08D \
                26A0 BB01 9700 E6B1 2976 4B92 BFC7 B96A 229B 73C6"
```

The license file includes the following sections:

- Header: General information
- General license: Shows the capacity licensed for the system, in this case 200 (TB).

Licensing error messages

The following table lists error messages that may be generated by the system, and their troubleshooting solutions.

Table 9 Licensing error messages

Error Message	Description	Solution
The license key is invalid or does not match this version. Contact Support.	The license key is invalid.	Contact support.
The current system configuration exceeds the license entitlements.	More capacity has been installed than the license allows.	Reduce capacity, or extend the license capacity.
Operation could not be completed. The license capacity has been exceeded.	When you try to add an SDS or device, it will cause the licensed capacity to be exceeded.	Do not add the SDS or device, or extend the license capacity.

Table 9 Licensing error messages (continued)

Error Message	Description	Solution
The license key is too long	The license file is larger than expected.	Check the accuracy of the license key.
The license has expired	The duration of the license has ended.	Extend the duration of the license.
The license installation ID does not match the ID of this system	When the Installation ID was entered in the ELM, it may have been incorrect.	Contact support.
The license contains a mismatch of the SWID. Contact Support.	The license key is invalid.	Contact support.
The issuer of the license you are attempting to add does not match that of the product	The license key is invalid.	Contact support.
The license contains a mismatch of the capacity values for basic and advanced features. Contact Support.	The capacity licensed for basic features is not equal to the capacity licensed for advanced feature.	Contact support.