Red Hat OpenShift Container Platform 4.6 on Dell Infrastructure

Enabled by Intel-powered Dell PowerEdge Servers; PowerSwitch Networking; and PowerMax, PowerScale, PowerStore, and Unity XT Storage

January 2022

H18955.1

Design Guide

Abstract

This design guide provides architecture and design information for the Dell Technologies Validated Design for Red Hat® OpenShift® Container Platform 4.6, for deployment on Intel-powered Dell PowerEdge servers, Dell PowerSwitch networking, and Dell PowerMax, PowerScale, PowerStore, and Unity XT storage systems.

Dell Technologies Solutions

Dell Technologies
Validated Design

Copyright

Contents

Chapter 1 Introduction 5
Solution overview 6
Document purpose 7
Audience 8
We value your feedback 8

Chapter 2 Technology and Deployment Process Overview 9
Introduction 10
Red Hat OpenShift Container Platform 10
Cloud-native infrastructure 13
Server options 18
Validated firmware 19
Deployment process 20
Infrastructure requirements 22

Chapter 3 Networking Infrastructure and Configuration 25
Introduction 26
OpenShift network operations 26
Physical network design 29

Chapter 4 Storage Overview 34
OpenShift Container Platform storage 35
OpenShift Data Foundation 38
Dell storage options 40
CSI external storage 41
Data protection 43

Chapter 5 Cluster Hardware Design 44
Introduction 45
Cluster scaling 45
Cluster hardware planning 45
Validated hardware configuration options 47

Chapter 6 Use Cases 51
Overview 52
Enterprise applications 52
Networking 55
Data analytics and artificial intelligence 57

Chapter 7 References **60**
Dell Technologies documentation 61
Red Hat documentation 61
Other resources 62

Appendix A Hardware Configuration **63**
Overview 64
Dell PowerEdge R640 node BOM 64
Dell PowerEdge R740xd node BOM 65
Dell PowerEdge R650 BOM 66
Dell PowerEdge R750 BOM 67
Dell PowerStore 1000T BOM 67
Dell Unity 380F BOM 69
Dell PowerMax BOM 69
OpenShift Data Foundation data node configurations 70

Chapter 1 Introduction

This chapter presents the following topics:

Solution overview 6

Document purpose 7

Audience 8

We value your feedback 8

Solution overview

Introduction

Red Hat® OpenShift® Container Platform 4.6 can host the development and run-time execution of containerized applications. OpenShift Container Platform is based on Kubernetes, the de facto automation and life cycle management (LCM) platform for containerized workloads and services.

The Dell Technologies Validated Design (DTVD) for OpenShift Container Platform 4.6 is a flexible infrastructure solution that has been designed, optimized, and validated for an OpenShift Container Platform 4.6 on-premises bare-metal deployment. The DTVD includes Dell servers, switches, and storage to enable you to develop, validate, and deploy your containerized applications. The emphasis is on running cloud-native applications on containers that are managed by OpenShift Container Platform.

Key benefits

The DTVD provides:

- A proven design to help organizations accelerate their container deployments and cloud-native adoption.
- A selection of validated OpenShift Container Platform hardware designs
- A scalable hardware platform of up to 210 compute nodes spread across seven racks
- Rapid implementation and reduced time to value

New features

This version of the DTVD for Red Hat OpenShift Container Platform 4.6 on Dell infrastructure includes the following new features:

Table 1. Change history

Publication date	New features
January 2022	Support for: • Red Hat Service Mesh
October 2021	Support for: • PowerEdge R650, R750 servers, based on 3rd Generation Intel® Xeon® Scalable Processors • PowerStore storage array, providing enterprise-grade FC, iSCSI, and NFS storage • NVIDIA V100 Graphical Processing Unit (GPU) on the Dell PowerEdge 740xd server

Solution documentation

The solution documentation set consists of:

- *Red Hat OpenShift Container Platform 4.6 on Dell EMC Infrastructure Implementation Guide*: Information about automation-assisted deployment of the solution.
- *Red Hat OpenShift Container Platform 4.6 on Dell EMC Infrastructure Design Guide*: This document.

The following OpenShift Container Platform 4.6 documents provide information about:

- Manual installation and deployment of Red Hat software products
- Platform deployment preferences OpenShift Container Platform 4.6 consists of many open-source components that have been carefully integrated to provide a consistently dependable platform on which you can develop and deploy scalable containerized applications.

Note: While you can rely on Red Hat Enterprise Linux security and container technologies to prevent intrusions and protect your data, some security vulnerabilities might persist. For information about security vulnerabilities in OpenShift Container Platform, see OCP Errata. For a general listing of Red Hat vulnerabilities, see the Red Hat Security Home Page.

Document purpose

This guide provides information for building an on-premises infrastructure solution to host OpenShift Container Platform 4.6. The guide describes the design decisions and configurations that were made to enable solution architects to:

- Design and deploy a container platform solution.
- Extend or modify the design as necessary to meet customer requirements.

This guide includes:

- Container ecosystem design overview
- Network infrastructure design guidance
- Container and application storage design guidance
- Server requirements to support OpenShift Container Platform node roles
- Configuration recommendations to optimize latency and networking performance
- General guidance for remote compute nodes at the edge
- Hardware platform configuration recommendations
- Rack-level design and power configuration considerations

The guide also provides information about:

- Red Hat OpenShift Container Platform 4.6 for application development and deployment
- Dell PowerEdge R640 and R740xd servers for compute and storage

- Dell PowerEdge R650 and R750 servers based on Intel® 3rd Generation Xeon® Processors
- Dell PowerEdge XE2420 servers for compute at the edge
- Dell PowerSwitch S5200-series switches for infrastructure network enablement
- Dell PowerSwitch S3048 switch for out-of-band (OOB) management of the cluster
- NVIDIA V100 GPUs on Dell PowerEdge R740xd servers

Audience

This design guide is for system administrators and system architects. Some experience with containers, Kubernetes, and the OpenShift Container Platform is recommended.

We value your feedback

Dell Technologies and the authors of this document welcome your feedback on the solution and the solution documentation. Contact the Dell Technologies Solutions team by email with your comments.

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

Chapter 2 Technology and Deployment Process Overview

This chapter presents the following topics:

Introduction 10

Red Hat OpenShift Container Platform 10

Cloud-native infrastructure 13

Server options 18

Validated firmware 19

Deployment process 20

Infrastructure requirements 22

Introduction

Overview

This chapter describes the OpenShift Container Platform architecture, infrastructure components, and requirements of a viable architecture for a Red Hat OpenShift Container Platform 4.6 cluster that can drive the core of modern telecommunications practices, multimedia operations, service provider infrastructure operations, and the demands of the gaming industry, enterprise workloads, and financial transaction workloads. This chapter also describes the Dell Technologies simplified bootstrapping process for an OpenShift Container Platform 4.6 cluster deployment.

Red Hat OpenShift Container Platform

Overview

OpenShift Container Platform is an enterprise-grade declarative state machine that has been designed to automate application workload operations based on the upstream Kubernetes project. In a Kubernetes context, “declarative” means that developers can specify, in code, a configuration for an application or workload without knowing how that application is going to be deployed. OpenShift Container Platform uses the OpenShift Kubernetes Engine, an enterprise-grade Kubernetes distribution, to provide production-oriented container and workload automation. OpenShift Container Platform 4.6 is based on Kubernetes version 1.19, which includes native support for cluster snapshots, enabling cluster backup and recovery. Also, OpenShift Container Platform 4.6 is the first extended user support release of OpenShift Container Platform, providing eighteen months of support. Built on top of Kubernetes, OpenShift Container Platform gives administrators and developers the tools that they need to deploy and manage applications and services at scale.

Note: OpenShift Container Platform is a certified Kubernetes distribution. Certification for Kubernetes distributions is provided by the Cloud Native Computing Foundation.

The following figure shows the OpenShift Container Platform architecture:

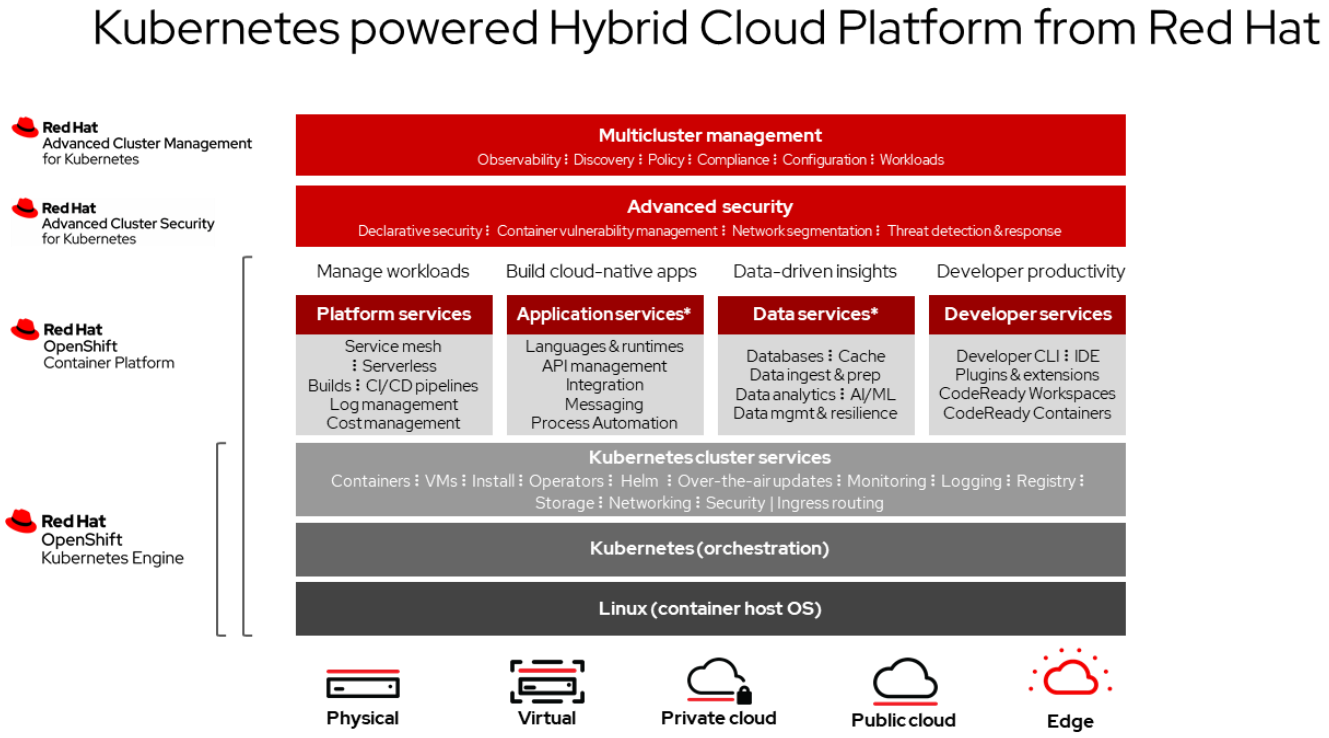


Figure 1. OpenShift Container Platform architecture

Kubernetes concepts

In Kubernetes, everything is an object. Every object has a current state, a preferred state, and a specification of how a state transition can be achieved. This specification includes everything from applications, deployments, and services to machine configuration and management of specific hardware resources. When a Kubernetes object is created, the cluster uses the object to transition towards the preferred state for the cluster.

Custom Resource Definitions (CRDs) can be used to specify new resource types, which can then be used to create Custom Resources (CRs). Middleware (typically, operators) can use this extensible mechanism to create resource types that Kubernetes and other middleware with appropriate access can manage and use.

What Kubernetes is

Kubernetes provides an abstraction layer for application containers, deployments, and services and automates all container operations.

Developers and administrators specify the needs of an application in a declarative manner, and Kubernetes automatically deploys, terminates, or restarts containers to converge on this preferred state.

What Kubernetes is not

There is no imperative management of containers in Kubernetes. Rather, Kubernetes consists of independent control processes (state transition machines) that move the current state of the cluster towards the preferred state. This mechanism has fundamental

implications for how cluster operations, application middleware, and more can be managed automatically (see Cluster automation).

Upstream Kubernetes has some fundamental limitations in that it does not build or deploy applications; provide logging, monitoring, or alerting mechanisms; and is not a self-healing, self-managing system. As an open-source project, Kubernetes must support various use cases and enable customers to use a wide variety of projects that are compatible with Kubernetes.

Why OpenShift Container Platform?

OpenShift Container Platform fills the gaps that Kubernetes leaves open, such as:

- Platform-level services, including building and packaging applications
- Integrated logging and monitoring solutions (Prometheus and Grafana)
- Integrated web console

OpenShift Container Platform is intended as a turnkey solution for production-grade environments. Among other benefits that it provides, OpenShift Container Platform:

- Eliminates the complexity of installing Kubernetes and adding authentication, management, logging, security, and networking.
- Provides additional self-management capabilities that are not found in Kubernetes due to the tightly coupled toolchain: the default containers-first operating system (Red Hat CoreOS), a Kubernetes-first container run-time (CRI-O), and a rigorous testing and certification process for additional Red Hat and vendor middleware.

Cluster automation

The operator framework gives vendors the ability to manage the life cycle of the middleware they provide—for example, the Dell CSI operator provides drivers for Dell storage products. Operators attempt to encode the operational knowledge that is required for various stateful applications. Depending on its complexity, the operator can be used to fully automate the life cycle management of various applications and middleware, automatically scale, and handle abnormalities gracefully. Like Helm, a universal package manager for Kubernetes, an operator can be used to configure and install middleware. Unlike Helm, operators are application-specific—an operator must be installed to manage each middleware application.

Operators are designed to simplify Day-2 operations by automatically deploying, updating, and maintaining specific application deployments. This simplification is achieved through the creation of CRDs that are managed through a control loop that is embedded in the operator.

The following figure shows the benefits that operators can provide, depending on their complexity:

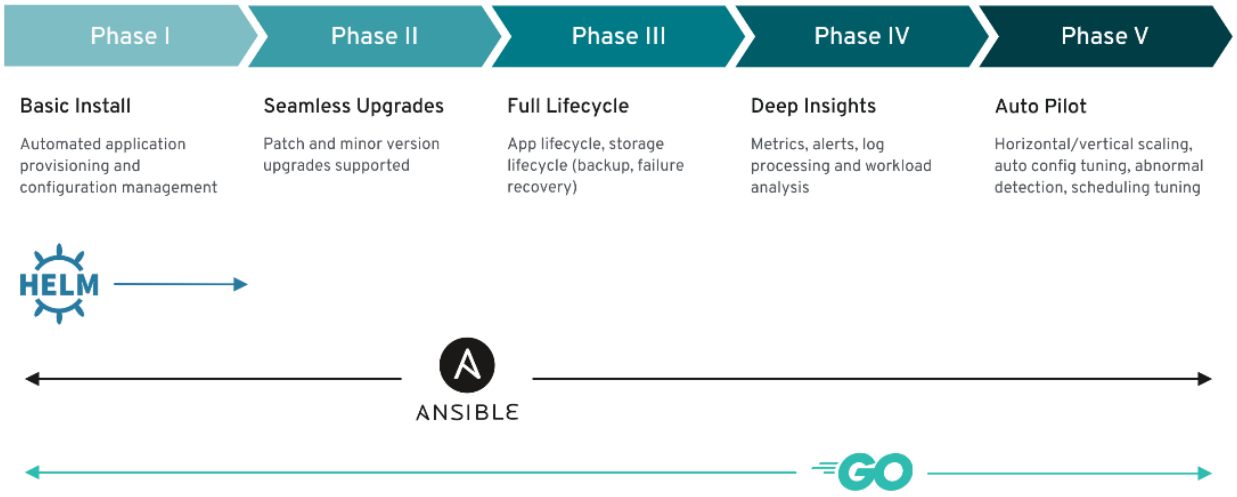
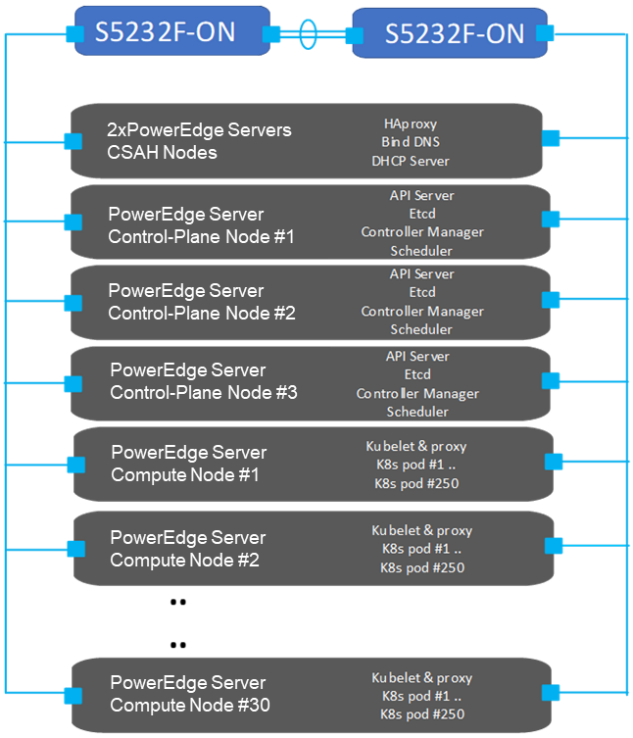


Figure 2

Cloud-native infrastructure

A cloud-native infrastructure must accommodate a large, scalable mix of service-oriented applications and their dependent components. These applications and components are generally microservices-based. The key to sustaining their operation is to have the right platform infrastructure and a sustainable management and control plane. The reference design that this guide describes helps you specify infrastructure requirements for building an on-premises OpenShift Container Platform 4.6 solution. The following figure shows this design:



OpenShift Container Platform 4.6 standard cluster design

Figure 3.

OpenShift Container Platform terminology

This architectural design recognizes four host types that make up every OpenShift Container Platform cluster: the bootstrap node, the control-plane nodes, the compute nodes, and the storage nodes.

The deployment process also requires a node called the Cluster System Admin Host (CSAH) node. For a description of the deployment process, see the *Red Hat OpenShift Container Platform 4.6 on Dell EMC Infrastructure Implementation Guide*.

Note: Red Hat official documentation does not refer to a CSAH node in the deployment process.

CSAH nodes

The CSAH nodes are not part of the cluster but they are required for OpenShift cluster administration and operation. CSAH nodes also provision DHCP, PXE, DNS, and HAProxy services for cluster operation. While a single CSAH node can be used for development and testing purposes, this approach does not provide resilient load-balancing. For resilient load-balancing, Dell Technologies recommends using two CSAH nodes running HAProxy and KeepAlived. Further, Dell Technologies strongly discourages directly logging in to a control-plane node to manage the cluster. The OpenShift CLI tool called `oc` and the authentication tokens that are required to administer the OpenShift cluster are installed on both CSAH nodes as part of the deployment process. For redundancy, it is recommended that you store backups of OpenShift authentication credentials outside the cluster.

Note: Control-plane nodes are deployed using immutable infrastructure, further driving the preference for an administration host that is external to the cluster.

Bootstrap node (VM)

The CSAH nodes manage the operation and installation of the container ecosystem cluster. Installation of the cluster begins with the creation of a bootstrap VM on the primary CSAH node to be used to install control-plane components on the nodes. The initial minimum cluster can consist of three nodes running both the control plane and applications, or three control-plane nodes and at least two compute nodes. OpenShift Container Platform requires three control-plane nodes in both scenarios.

Basic node configuration

Node components are installed and run on every node in the cluster; that is, on control-plane nodes and compute nodes. The components are responsible for all node run-time operations. Key components consist of:

- **Kubelet:** An agent that runs on each node to perform declarations or actions that are provided to the cluster-API. Kubelet performs node service functions to ensure that running pods are compliant with PodSpecs and remain healthy. Kubelet does not manage containers or pods that were not created by Kubernetes.
- **Kube-proxy:** An instance of kube-proxy runs on every node of the cluster. The kube-proxy instance implements Kubernetes network services that run on the node. Kube-proxy also manages network connectivity and traffic route management based on host operating system packet filtering.
- **Container run-time:** The chosen container run-time engine (CRE) must be deployed on each node in a Kubernetes cluster. The CRE must comply with the Kubernetes Container Runtime Interface (CRI) specifications. OpenShift Container Platform defaults to the CRI-O container run-time and cannot be changed.

Control-plane nodes

Nodes that implement control-plane infrastructure management are called control-plane nodes. Three control-plane nodes establish a unified control plane for the operation of an OpenShift cluster. The control plane operates outside the application container workloads and is responsible for ensuring the overall continued viability, health, availability, and integrity of the container ecosystem. Removing control-plane nodes is not allowed.

OpenShift Container Platform also deploys additional control-plane infrastructure to manage OpenShift-specific cluster components.

The control plane provides the following functions:

- **API Server:** The API server exposes the Kubernetes control-plane API for other platform services (such as a web console) to consume and has API endpoints to manage cluster resources.
- **Etcd:** A highly available and consistent key-value store used to maintain Kubernetes cluster data. The etcd daemon is run on each control plane node and requires a majority consensus to achieve quorum (the formula used for quorum is $\frac{n}{2} + 1$, where *n* is the number of control plane nodes). For production clusters, at least three control-plane nodes are required, each running an etcd daemon. This requirement means that at least two control planes are required to achieve quorum.
- **Scheduler:** The Kubernetes scheduler assigns new pods to a node based on the resource requirements (for CPU, RAM, and GPU, for example) and the affinity and anti-affinity mechanisms.

- **Controller manager:** The controller managers run all controller processes. While each controller process is independent, the processes are run as a single executable to reduce complexity. The controllers include the node, replication, endpoints, service, and token controllers.
- **OpenShift API server:** The OpenShift API server validates and configures the data for OpenShift resources such as projects, routes, and templates. The OpenShift API server is managed by the OpenShift API server operator.
- **OpenShift controller manager:** The OpenShift controller manager watches etcd for changes to OpenShift objects such as project, route, and template controller objects, and then uses the API to enforce the specified state. The OpenShift controller manager is managed by the OpenShift controller manager operator.
- **OpenShift OAuth API server:** The OpenShift OAuth API server validates and configures the data to authenticate to OpenShift Container Platform, such as users, groups, and OAuth tokens. The OpenShift OAuth API server is managed by the cluster authentication operator.
- **OpenShift OAuth server:** Users request tokens from the OpenShift OAuth server to authenticate themselves to the API. The OpenShift OAuth server is managed by the cluster authentication operator.

Backup and disaster recovery

Even though OpenShift Container Platform is resilient to node failure, it is recommended to take regular backups of the etcd data store. Because etcd backups are a blocking procedure, take them at off-peak hours in production environments. Keep in mind that when you update a cluster within minor versions (for example, from 4.6.2 to 4.6.3), you should take an etcd backup of the version of OpenShift Container Platform that is currently running on your cluster or clusters. Take etcd backups 24 hours after the cluster has been installed to let the initial rotation of certificates occur; otherwise, the etcd backup may contain expired certificates. For more information, see the Red Hat document Backing up etcd.

Quorum requirements for etcd dictate that if enough control-plane nodes fail (and, as a result, most control planes are no longer operating), restoring from a previous cluster state becomes the only option for cluster recovery. For the cluster restoration steps, see Restoring to a previous cluster state. If most control-plane nodes are still operating, meaning that quorum can be achieved but no redundancy exists for further node failure, it is necessary to replace unhealthy etcd members. To perform this task, follow the steps in Replacing an unhealthy etcd member.

Compute plane

In an OpenShift cluster, application containers are deployed to run on compute nodes by default. The term “compute node” is arbitrary; nothing specific is required to run compute nodes, and applications can be run on control-plane nodes, if wanted. Cluster nodes advertise their resources and resource utilization so that the scheduler can allocate containers and pods to these nodes and maintain a reasonable workload distribution. The Kubelet service runs on all nodes in a Kubernetes cluster. This service receives container deployment requests and ensures that the requests are instantiated and put into operation. The Kubelet service also starts and stops container workloads and manages a service proxy that handles communication between pods that are running across compute nodes.

Logical constructs called MachineSets define compute node resources. MachineSets can be used to match requirements for a pod deployment to a matching compute node. OpenShift Container Platform supports defining multiple machine types, each of which defines a compute node target type.

Compute nodes can be added to or deleted from a cluster if doing so does not compromise the viability of the cluster. If the control-plane nodes are not designated as schedulable, at least two viable compute nodes must always be operating to run router pods that manage ingress networking traffic. Further, enough compute platform resources must be available to sustain the overall cluster application container workload.

Storage nodes

Storage can be either provisioned from dedicated nodes or shared with compute services. Provisioning occurs on disk drives that are locally attached to servers that have been added to the cluster as compute nodes. For more information, see Dell storage options.

The Red Hat Data Services portfolio of solutions includes persistent software-defined storage (SDS) and data services that are integrated with and optimized for OpenShift Container Platform. As part of the portfolio, Red Hat OpenShift Data Foundation (formerly known as OpenShift Container Storage) delivers resilient and persistent SDS and data services based on Ceph, Rook, and NooBaa technologies.

OpenShift Data Foundation

Running as a Kubernetes service, OpenShift Data Foundation is engineered, tested, and certified to provide data services for OpenShift Container Platform on any infrastructure. OpenShift Data Foundation can be deployed within an OpenShift Container Platform cluster on existing worker nodes, infrastructure nodes, or dedicated nodes. Alternatively, OpenShift Data Foundation can be decoupled and managed as a separate, independently scalable data store, delivering data for one or many OpenShift Container Platform clusters. To streamline configuration options, Red Hat and Intel® have jointly developed three workload-optimized configurations for OpenShift Data Foundation external data nodes: edge, capacity, and performance. These configurations are optimized for Dell PowerEdge R750 servers, as described in Appendix A. It is also possible to use existing compute nodes if they meet OpenShift Data Foundation hardware requirements.

You can start the deployment of OpenShift Data Foundation from the embedded OperatorHub when you are logged into OpenShift Container Platform as the cluster administrator. For more information, see OpenShift Container Storage 4.6 Documentation.

Note: At the time of publication of this guide, some Red Hat documentation and the operator and product interface of OpenShift Data Foundation may still use the product name OpenShift Container Storage for OpenShift Data Foundation.

Server options

Dell Technologies has tested and fully validated the following PowerEdge server options for an OpenShift Container Platform 4.6 deployment:

PowerEdge R650 server

The PowerEdge R650 server, which is powered by the 3rd Generation Intel® Xeon® Scalable processors, is the optimal rack server to address application performance and acceleration. The PowerEdge R650 is a two-socket/1U rack server that delivers outstanding performance for the most demanding workloads. The PowerEdge R650 supports eight channels of memory per CPU, and up to 32 DDR4 DIMMs at 3200 MT/s speeds. To address substantial throughput improvements, the PowerEdge R650 model supports PCIe Gen 4 and up to 10 NVMe drives with improved air-cooling features plus optional Direct Liquid Cooling to support increasing power and thermal requirements. This makes the PowerEdge R650 an ideal server for data center standardization on a wide range of workloads.

PowerEdge R750 server

The PowerEdge R750 server, which is powered by the 3rd Generation Intel® Xeon® Scalable processors, is a rack server to address application performance and acceleration. The PowerEdge R750 is a dual-socket/2U rack server that delivers outstanding performance for the most demanding workloads. The PowerEdge R750 supports eight channels of memory per CPU and up to 32 DDR4 DIMMs at 3200 MT/s speeds. To address substantial throughput improvements, the PowerEdge R750 supports PCIe Gen 4 and up to 24 NVMe drives with improved air-cooling features plus optional Direct Liquid Cooling to support increasing power and thermal requirements. The PowerEdge R750 is therefore an ideal server for data center standardization on a wide range of workloads requiring performance, extensive storage, and GPU support.

PowerEdge R640 server

The PowerEdge R640 server is the ideal two-socket, 1U platform for dense scale-out cloud computing. The scalable business architecture of the PowerEdge R640 platform is designed to maximize application performance and provide you with the flexibility to optimize configurations based on the application and use case. With the PowerEdge R640, you can use up to eight NVMe drives, or use either 2.5” or 3.5” drives for data storage. Combined with up to 24 DIMMs, 12 of which can be NVDIMMs, you have the resources to create the optimum configuration to maximize application performance in only a 1U chassis.

PowerEdge R740xd server

The PowerEdge R740xd server delivers a perfect balance between storage scalability and performance. The 2U two-socket platform is ideal for software-defined storage (SDS), supporting up to 24 NVMe, or 32 x 2.5”, or 18 x 3.5” drives. The versatility of the R740xd model is shown by its ability to mix any drive type to create the optimum configuration of NVMe, SATA SSD, and hard drive for performance, capacity, or both.

PowerEdge XE2420 server

The PowerEdge XE2420 server is a configurable, two-socket, 2U rack server that delivers powerful 2S performance in a short-depth form-factor. With a scalable rack option, the XE2420t model is ideal for low-latency, large storage edge applications. Its performance can be further enhanced by its support of up to four accelerators. A wide and flexible range of storage options (both SSD and NVMe) and a large capacity, of up to 132 TB, gives it the flexibility to tackle a diverse set of demanding workloads at the edge. The

PowerEdge XE2420 is a specialty edge server that is engineered to deliver powerful performance for harsh environments with its ability to perform under extended operating temperatures from 5°C to 45°C. It is a front-accessible server that is designed to support demanding edge applications such as streaming analytics, manufacturing logistics, and 5G cell processing.

Validated firmware

The following tables show the firmware that Dell Technologies has validated for an OpenShift Container Platform 4.6 deployment using the PowerEdge R640 and R740xd, the XE2420, and the R650 and R750 server platforms respectively:

Table 2. PowerEdge R640 and R740xd tested firmware

Product	Version
BIOS	2.10.2
iDRAC with Lifecycle Controller	4.40.10.00
Mellanox ConnectX-4	14.28.45.12
Mellanox ConnectX-5 EX	16.28.45.12
Intel® XXV710	20.0.17
BOSS-S1	2.5.13.3024
HBA330 Mini	16.17.01.00
PERC H730P Mini controller	25.5.8.0001

Table 3. PowerEdge XE2420 tested firmware

Product	Version
BIOS	1.4.1
iDRAC with Lifecycle Controller	4.40.10.00
Intel® XXV710	19.5.12
BOSS-S1	2.5.13.3024
HBA330 Mini	16.17.01.00

Table 4. PowerEdge R650 and R750 tested firmware

Product	Version
BIOS	1.2.5
iDRAC with Lifecycle Controller	5.10.00.00
Mellanox ConnectX-5 25G	16.31.20.06
PERC H755N Front	52.16.1-4158
BOSS-S2	2.5.13.4008

Deployment process

Dell Technologies has simplified the process of bootstrapping the OpenShift Container Platform 4.6 cluster. To use the simplified process, ensure that:

- The cluster is provisioned with network switches and servers.
- Network cabling is complete.
- Internet connectivity has been provided to the cluster. Internet connectivity is necessary to install OpenShift Container Platform 4.6.

The deployment begins with initial switch provisioning. Initial switch provisioning enables preparation and installation of the CSAH node and consists of:

- Installing Red Hat Enterprise Linux 8
- Subscribing to the necessary repositories
- Creating an Ansible user account
- Cloning a GitHub Ansible playbook repository from the Dell ESG container repository
- Running an Ansible playbook to initiate the installation process

Dell Technologies has generated Ansible playbooks that fully prepare both CSAH nodes. Before the installation of the OpenShift Container Platform 4.6 cluster begins, the Ansible playbook sets up a PXE server, DHCP server, DNS server, HAProxy, and HTTP server. If a second CSAH node is deployed, the playbook also sets up DNS, HAProxy, HTTP, and KeepAlived services on that node. The playbook creates ignition files to drive installation of the bootstrap, control-plane, and compute nodes, and it also starts the bootstrap VM to initialize control-plane components. The playbook presents a list of node types that must be deployed in top-down order.

Note: For enterprise sites, consider deploying appropriately hardened DHCP and DNS servers and using resilient multiple-node HAProxy configuration. The Ansible playbook for this design can deploy multiple CSAH nodes for resilient HAProxy configuration. This guide provides CSAH Ansible playbooks for reference only at the implementation stage.

The Ansible playbook creates an `install-config.yaml` file that is used to control deployment of the ***bootstrap*** node. For more information, see the *Red Hat OpenShift Container Platform 4.6 on Dell EMC Infrastructure Implementation Guide*.

An ignition configuration control file starts the bootstrap node, as shown in the following figure:

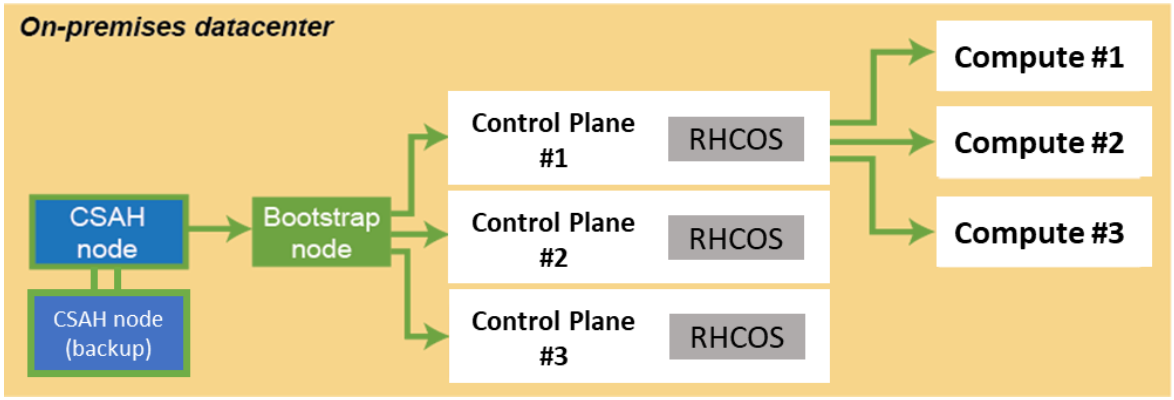


Figure 4. **Installation workflow: Creating the bootstrap, control-plane, and compute nodes**

Note: An installation that is driven by ignition configuration generates security certificates that expire after 24 hours. You must install the cluster before the certificates expire, and the cluster must operate in a viable (nondegraded) state so that the first certificate rotation can be completed.

The cluster bootstrapping process consists of the following phases:

1. After startup, the bootstrap VM creates the resources that are required to start the control-plane nodes. Interrupting this process can prevent the OpenShift control plane from coming up.
2. The control-plane nodes pull resource information from the bootstrap VM to bring them up to a viable state. This resource information is used to form the etcd control plane cluster.
3. The bootstrap VM instantiates a temporary Kubernetes control plane that is under etcd control.
4. A temporary control plane loads the application workload control plane to the control-plane nodes.
5. The temporary control plane is shut down, handing control over to the now viable control-plane nodes.
6. OpenShift Container Platform components are pulled into the control of the control-plane nodes.
7. The bootstrap VM is shut down.
8. The control-plane nodes now drive creation and instantiation of the compute nodes.
9. The control plane adds operator-based services to complete the deployment of the OpenShift Container Platform ecosystem.

The cluster is now viable and can be placed into service in readiness for Day-2 operations. You can expand the cluster by adding more compute nodes to suit your requirements.

Infrastructure requirements

Cluster sizing

In OpenShift Container Platform 4.6, two different types of cluster deployments are available: a three-node cluster and a standard cluster (more than five nodes). In a three-node cluster, the control plane and cluster workloads are run on the same nodes, allowing for small footprint deployments of OpenShift for testing, development, and production environments. While the three-node cluster can be expanded with additional compute nodes, an initial expansion of a three-node cluster requires that at least two compute nodes be added simultaneously. This is because the ingress networking controller deploys two router pods on compute nodes for full functionality. If compute nodes are added to a three-node cluster, deploy two compute nodes for full ingress functionality. You can add compute nodes subsequently as needed. A standard cluster deployment has three control-plane nodes and at least two compute nodes. With this deployment type, control-plane nodes are marked as “unschedulable,” preventing cluster workloads from being scheduled on those nodes. Both cluster deployment types require two CSAH nodes for cluster management and resilient load-balancing.

Cluster infrastructure guidance

The following table provides basic cluster infrastructure guidance for validated hardware configurations. For detailed configuration information, see Cluster Hardware Design. A container cluster can be deployed quickly and reliably when each node is within the validated design guidelines.

Table 5. Hardware infrastructure for OpenShift Container Platform 4.6 cluster deployment

Type	Description	Count	Notes
CSAH node	Dell PowerEdge R640/R650 server	2	Creates a bootstrap VM. CSAH runs a single instance of HAProxy. For an enterprise high availability (HA) deployment of OpenShift Container Platform 4.6, Dell Technologies recommends using a commercially supported L4 load-balancer or proxy service, or an additional PowerEdge R640 CSAH node running HAProxy and KeepAlived alongside the primary CSAH node. Options include commercial HAProxy, Nginx, and F5.
Control-plane nodes	PowerEdge R640/R650 server	3	Deployed using the bootstrap VM.
Compute nodes	PowerEdge R640 or R740xd server PowerEdge R650 or R750 server	A minimum of 2* per rack, maximum 30	No compute nodes are required for a three-node cluster. A standard deployment requires a minimum of two compute nodes (and three controller nodes). To expand a three-node cluster, you must add two compute nodes simultaneously. After the cluster is operational, you can add more compute nodes to the cluster through the Cluster Management Service.

Type	Description	Count	Notes
Data switches	Either of the following switches: • Dell PowerSwitch S5248-ON • Dell PowerSwitch S5232-ON	2 per rack	Configured at installation time. **Note:** • HA network configuration requires two data path switches per rack. • Multirack clusters require network topology planning. Leaf-spine network switch configuration may be necessary.
iDRAC network	Dell PowerSwitch S3048-ON	1 per rack	Used for OOB management.
Rack	Selected according to site standards	1–3 racks	For multirack configurations, consult your Dell Technologies or Red Hat representative regarding custom engineering design.

*A three-node cluster does not require any compute nodes. To expand a three-node cluster with additional compute machines, you must first expand the cluster to a five-node cluster using two additional compute nodes.

Minimum viable solution requirements

Installing OpenShift Container Platform requires, at a minimum, the following nodes:

- One CSAH node, which is used to run the bootstrap VM. While the cluster is in production use, the CSAH node manages the cluster and performs load-balancing for the cluster. Optionally, if resilient load-balancing is required, a second CSAH node can be used with the primary CSAH node to perform highly available load-balancing.
- Three nodes running both the control plane and data plane, enabling customers to deploy OpenShift Container Platform 4.6 using only four nodes.

HA of the key services that make up the OpenShift Container Platform cluster is necessary to ensure run-time integrity. Redundancy of physical nodes for each cluster node type is an important aspect of HA for the bare-metal cluster.

In this design guide, HA includes the provisioning of at least two network interface controllers (NICs) and two network switches that are configured to provide redundant pathing. The redundant pathing provides for network continuity if a NIC or a network switch fails. HA load-balancing can be provided by using an enterprise-grade load balancer or an additional PowerEdge R640 server running HAProxy and KeepAlived alongside the CSAH node.

OpenShift Container Platform 4.6 must use Red Hat Enterprise Linux CoreOS 4.6 (RHCOS) for the control-plane nodes and compute nodes.

Note: Using Red Hat Enterprise Linux 7 compute nodes is now deprecated and the ability to use them in OpenShift will be removed in a future release of OpenShift. For that reason, this design guide and its companion implementation guide no longer leverage Red Hat Enterprise Linux 7 compute nodes. The bootstrap and control-plane nodes must use RHCOS 4.6 as their operating system. Each of these nodes must be immutable.

The following table shows the minimum resource requirements for the nodes:

Table 6. Minimum resource requirements for OpenShift Container Platform 4.6 nodes

Node type	Operating system	Minimum CPU cores	RAM	Storage
CSAH	Red Hat Enterprise Linux 8.2	4	32 GB	200 GB
Bootstrap	RHCOS 4.6	4	16 GB	120 GB
Controller	RHCOS 4.6	4	16 GB	120 GB
Compute	RHCOS 4.6	2	8 GB	120 GB

Network connectivity requirements

The RHCOS nodes must fetch ignition files from the Machine Config server. This operation uses an `initramfs-based-node` startup for the initial network configuration. The startup requires a DHCP server to provide a network connection that gives access to the ignition files for that node. Subsequent operations can use static IP addresses.

Chapter 3 Networking Infrastructure and Configuration

This chapter presents the following topics:

Introduction 26

OpenShift network operations 26

Physical network design 29

Introduction

The components and operations that make up the container ecosystem each require network connectivity plus the ability to communicate with all other components and respond to incoming network requests. The reference design that this guide describes uses Dell PowerSwitch networking infrastructure.

OpenShift network operations

Operating components

Applications run on compute nodes. Each compute node is equipped with resources such as CPU cores, memory, storage, NICs, and add-in host adapters including GPUs, SmartNICs, and FPGAs. Kubernetes provides a mechanism to enable orchestration of network resources through the Container Network Interface (CNI) API.

The CNI API uses the Multus CNI plug-in to enable attachment of multiple adapter interfaces on each pod. CRD objects are responsible for configuring Multus CNI plug-ins.

Container communications

A pod, which is a basic unit of application deployment, consists of one or more containers that are deployed together on the same compute node. A pod shares the compute node network infrastructure with the other network resources that make up the cluster. As service demand expands, additional identical pods are often deployed to the same or other compute nodes.

Networking is critical to the operation of an OpenShift Container cluster. Four basic network communication flows occur within every cluster:

- Container-to-container connections (also known as highly coupled communication)
- Pod communication over the local host network (127.0.0.1)
- Pod-to-pod connections, as described in this guide
- Pod-to-service and ingress-to-service connections, which are handled by services

Containers that communicate within their pod use the local host network address. Containers that communicate with any external pod originate their traffic based on the IP address of the pod.

Application containers use shared storage volumes (configured as part of the pod resource) that are mounted as part of the shared storage for each pod. Network traffic that might be associated with nonlocal storage must be able to route across node network infrastructure.

Services networking

Services are used to abstract access to Kubernetes pods. Every node in a Kubernetes cluster runs a kube-proxy and is responsible for implementing virtual IP (VIP) for services. Kubernetes supports two primary modes of finding (or resolving) a service:

- **Using environment variables:** This method requires a reboot of the pods when the IP address of the service changes.
- **Using DNS:** OpenShift Container Platform 4.6 uses CoreDNS to resolve service IP addresses.

Some parts of the application (front ends, for example) might need to expose a service outside the application. If the service uses HTTP, HTTPS, or any other TLS-encrypted protocol, use an ingress controller; for other protocols, use a load balancer, external service IP address, or node port.

A node port exposes the service on a static port on the node IP address. A service with `NodePort-type` as a resource exposes the resource on a specific port on all nodes in the cluster. Ensure that external IP addresses are routed to the nodes.

Ingress controller

OpenShift Container Platform uses an ingress controller to provide external access. The ingress controller defaults to running on two compute nodes, but it can be scaled up as required. Dell Technologies recommends creating a wildcard DNS entry and then setting up an ingress controller. This method enables you to work only within the context of an ingress controller. An ingress controller accepts external HTTP, HTTPS, and TLS requests using SNI, and then proxies them based on the routes that are provisioned.

You can expose a service by creating a route and using the cluster IP. Cluster IP routes are created in the OpenShift Container Platform project, and a set of routes is admitted into ingress controllers.

You can perform sharding (horizontal partitioning of data) on route labels or name spaces. Sharding enables you to:

- Load-balance the incoming traffic.
- Hive off the required traffic to a single ingress controller.

Networking operators

The following operators are available for network administration:

- **Cluster Network Operator (CNO):** Deploys the OpenShift SDN plug-in during cluster installation and manages kube-proxy on each node.
- **DNS operator:** Deploys and manages CoreDNS and instructs pods to use the CoreDNS IP address for name resolution.
- **Ingress operator:** Enables external access to OpenShift Cluster Platform cluster services and deploys and manages one or more HAProxy-based ingress controllers to handle routing.

Container Networking Interface

The CNI specification serves to make the networking layer of containerized applications pluggable and extensible across container run-times. The specification is used in both upstream Kubernetes and OpenShift in the pod network. This use is not implemented by Kubernetes, but by various CNI plug-ins. The most commonly used CNI plug-ins are:

- **Multus:** CNI plug-in that supports the multinetwork function in Kubernetes. While Kubernetes pods typically have only one networking interface, the use of Multus means that pods can be configured to support multiple interfaces. Multus acts as a “meta plug-in,” a plug-in which calls other CNI plug-ins. Multus also supports SR-IOV and DPDK workloads.
- **DANM:** Developed by Nokia, DANM is a CNI plug-in for telecom-oriented workloads. DANM supports the provisioning of advanced IPVLAN interfaces, acts like Multus in that it is also a meta plug-in, can control VxLAN and VLAN interfaces

for all Kubernetes hosts, and more. The DANM CNI plug-in creates a network management API to give administrators greater control of the physical networking stack through the standard Kubernetes API.

OpenShift SDN

OpenShift SDN creates an overlay network that is based on Open Virtual Switch (OVS). The overlay network enables communication between pods across the cluster. OVS operates in one of the following modes:

- Network policy mode (the default), which allows custom isolation policies
- Multitenant mode, which provides project-level isolation for pods and services
- Subnet mode, which provides a flat network

OpenShift Container Platform 4.6 also supports using Open Virtual Network (OVN)-Kubernetes as the CNI network provider. OVN-Kubernetes will become the default CNI network provider in a future release of OpenShift Container Platform. OpenShift Container Platform 4.6 supports additional SDN orchestration and management plug-ins that comply with the CNI specification. See Use cases for examples.

Service Mesh

Distributed microservices work together to make up an application. Service Mesh provides a uniform method to connect, manage, and observe microservices-based applications. The Red Hat OpenShift implementation of Service Mesh is based on Istio, an open-source project. Use operators from the OperatorHub to install Service Mesh; OpenShift Service Mesh is not installed automatically as part of a default installation.

Service Mesh has key functional components that belong to either the data plane or the control plane:

- **Envoy proxy:** Intercepts all traffic for all services in Service Mesh. Envoy proxy is deployed as a sidecar.
- **Mixer:** Enforces access control and collects telemetry data.
- **Pilot:** Provides service discovery for the envoy sidecars.
- **Citadel:** Provides strong service-to-service and end-user authentication with integrated identity and credential management.

Service Mesh can be employed at the cluster level or the project level. Users define the granularity of the Service Mesh deployment, enabling them to meet their specific deployment and application needs. For more information, see OpenShift Service Mesh documentation.

Multinetwork support

OpenShift Container Platform 4.6 also supports software-defined multiple networks. The platform comes with a default network. The cluster administrator defines additional networks using the Multus CNI plug-in, and then chains the plug-ins. The additional networks are useful for increasing the networking capacity of the pods and meeting traffic separation requirements.

The following CNI plug-ins are available for creating additional networks:

- **Bridge:** The same host pods can communicate over a bridge-based additional network.

- **Host-device:** Pods can access the physical Ethernet network device of the host.
- **Macvlan:** Pods attached to a macvlan-based additional network have a unique MAC address and communicate using a physical network interface.
- **Ipvlan:** Pods communicate over an ipvlan-based additional network.

Leaf-switch considerations

When pods are provisioned with additional network interfaces that are based on macvlan or ipvlan, corresponding leaf-switch ports must match the VLAN configuration of the host. Failure to properly configure the ports results in a loss of traffic.

Physical network design

Design principles

Dell networking products are designed for ease of use and to enable resilient network creation. OpenShift Container Platform 4.6 introduces various advanced networking features to enable containers for high performance and monitoring. Dell Technologies recommends designs that apply the following principles:

- Meet network capacity and the separation requirements of the container pod.
- Configure dual-homing of the OpenShift Container Platform node to two Virtual Link Trunked (VLT) switches.
- Create a scalable and resilient network fabric to increase cluster size.
- Allows monitor and trace container communications.

Container network capacity and separation

Container networking takes advantage of the high speed (25/100 GbE) network interfaces of the Dell server portfolio. Also, to meet network capacity requirements, pods can attach to more networks by using available CNI plug-ins.

Additional networks are useful when network traffic isolation is required. Networking applications such as Container Network Functions (CNFs) have control traffic and data traffic. These different types of traffic have different processing, security, and performance requirements.

Pods can be attached to the SR-IOV virtual function (VF) interface on the host system for traffic isolation and to increase I/O performance.

Dual-homing

Dual-homing means that each node that makes up the OpenShift cluster has at least two NICs, each connected to at least two switches. The switches require VLT connections so that they operate together as a single unit of connectivity to provide redundant data paths for all network traffic. The NICs at each node and the ports that they connect to on each of the switches can be aggregated using bonding to assure HA operation.

Network fabric

A nonblocking fabric is required to meet the needs of the microservices data traffic. Dell Technologies recommends deploying a leaf-spine network.

Monitoring and tracing

OpenShift Container Platform 4.6 supports Service Mesh. Users can monitor container traffic by using Kiali and perform end-to-end tracing of applications by using Jaeger.

Resilient networking

Each server that has many NIC options in the rack is connected to:

- Two leaf switches with a network interface of choice: 10 GbE, 25 GbE, or 100 GbE
- A management switch (typically, 1 GbE) for iDRAC connectivity

Our network design employs a VLT connection between the two leaf switches. In a VLT environment, all paths are active; therefore, it is possible to achieve high throughput while still protecting against hardware failures.

VLT technology allows a server to uplink multiple physical trunks into more than one PowerSwitch switch by treating the uplinks as one logical trunk. A VLT-connected pair of switches acts as a single switch to a connecting server. Both links from the bridge network can forward and receive traffic. VLT replaces Spanning Tree Protocol (STP)-based networks by providing both redundancy and full bandwidth utilization using multiple active paths.

The major benefits of VLT technology are:

- Dual control plane for highly available, resilient network services
- Full utilization of the active link aggregation (LAG) interfaces
- Active/active design for seamless operations during maintenance events

The VLTi configuration in this design uses two 100 GbE ports between each ToR switch. The remaining 100 GbE ports can be used for high-speed connectivity to spine switches, or directly to the data center core network infrastructure.

Scale out with leaf-spine fabric

You can scale container solutions by adding multiple compute nodes and storage nodes. A container cluster can have multiple racks of servers. To create a nonblocking fabric that meets the needs of the microservices data traffic, we used a leaf-spine network.

Leaf-spine overview

Layer 2 and Layer 3 leaf-spine topologies employ the following concepts:

- Each leaf switch connects to every spine switch in the topology.
- Servers, storage arrays, edge routers, and similar devices connect to leaf switches, but never to spines.

Our design used dual leaf switches at the top of each rack. We employed VLT in the spine layer, which allows all connections to be active while also providing fault tolerance. As administrators add racks to the data center, leaf switches are added to each new rack.

The total number of leaf-spine connections is equal to the number of leaf switches multiplied by the number of spine switches. Administrators can increase the bandwidth of

the fabric by adding connections between leaf switches and spine switches if the spine layer has capacity for the additional connections.

Layer 3 leaf-spine network

In a Layer 3 leaf-spine network, traffic is routed between leaf switches and spine switches. The Layer 3-Layer 2 boundary is at the leaf switches. Spine switches are never connected to each other in a Layer 3 topology. Equal cost multipath routing (ECMP) is used to load-balance traffic across the Layer 3 network. Connections within racks from hosts to leaf switches are Layer 2. Connections to external networks are made from a pair of edge or border leaf switches, as shown in the following figure:

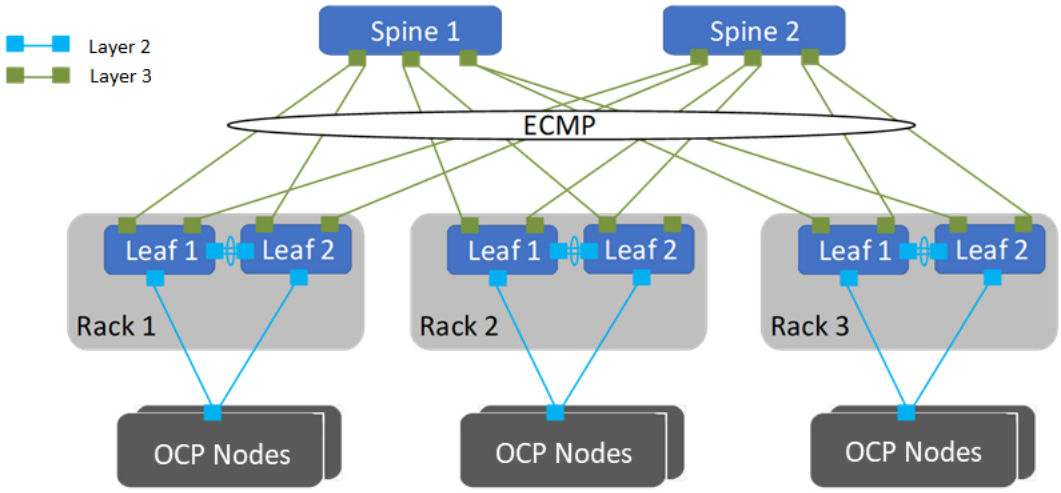


Figure 5. Leaf-spine network configuration

PowerSwitch configuration

Dell's high-capacity network switches are cost-effective and easy to deploy. The switches provide a clear path to a software-defined data center, offering:

- High density for 25, 40, 50, or 100 GbE deployments in top-of-rack (ToR), middle-of-row, and end-of-row deployments
- A choice of S5048F-ON, S5148F-ON, S5212F-ON, S5224F-ON, S5248F-ON, S5296F-ON, and S5232F-ON 25 GbE and 100 GbE switches; and the S6100-ON 10 GbE, 25 GbE, 40 GbE, 50 GbE, or 100 GbE modular switch
- S6100-ON modules that include: 16-port 40 GbE QSFP+; eight-port 100 GbE QSFP28; combo module with four 100 GbE CXP ports and four 100 GbE QSFP28 ports

For our solution design, we used Dell Network Operating System OS10. OS10 allows multilayered disaggregation of network functions that are layered on an open-source Linux-based operating system. The following section describes a high-level configuration of the PowerSwitch switches that are used for an OpenShift Container Platform deployment at various scales.

Configuring VLT

At a high level, the VLT configuration consists of the following steps:

1. Enable Spanning Tree (the default) on the VLT peer switches. Spanning Tree is recommended to prevent loops in a VLT doma-inch RPVST+ (the default) and RSTP modes are supported on VLT ports.
2. Create a VLT domain and configure the VLT interconnect (VLTi).
3. Configure the VLT Priority, VLT MAC Address, and VLT Backup Link.
4. Configure the LAG for the connected device.
5. Verify and monitor the status of VLT by using OS10 show commands.

Installation with Ansible

Dell networking modules are supported in Ansible core from Ansible 2.3 on. You can use these modules to manage and automate Dell switches running OS10. The modules are run in local connection mode using CLI and SSH transport.

For an example of a Clos fabric deployment based on the Border Gateway Protocol (BGP), see Provision CLOS fabric using Dell EMC Networking Ansible modules example.

High availability and load balancing

This solution uses the following HA features:

- **Red Hat OpenShift Container Platform 4.6:** Multiple control-plane nodes and infrastructure nodes
- **Resilient load balancing:** Two CSAH nodes running HAProxy and KeepAlived
- **Dell cloud-native infrastructure:** PowerEdge servers with dual NICs
- **Dell PowerSwitch:** Spine-leaf fabric with VLT

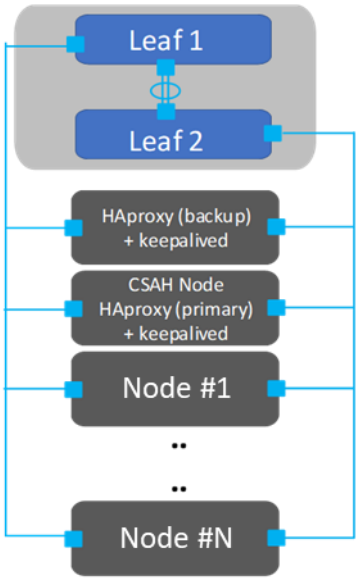
Keepalived

`Keepalived` is an open-source project that implements routing software using the Virtual Router Redundancy Protocol (VRRP). VRRP enables a switchover to a backup server if the primary server fails. This switchover is achieved by using VIP. To configure `keepalived` on both servers:

- Configure one server as "`MASTER`" with high priority. This will be the primary server.
- Configure the other server as "`BACKUP`" with lower priority.

Always make external traffic paths highly available to create a complete solution. The cluster administrator can use an external L4 load-balancer in a highly available manner or deploy HAProxy in resilient mode. Deploying a resilient HAProxy requires one additional server. As shown in the following figure, the components of a highly available load-balancer design using HAProxy are:

- **Keepalived and HAProxy running on CSAH:** Configure VIP on a suitable network interface.
- **Keepalived and HAProxy running on an additional server**: Configure VIP on a suitable network interface.



Highly available load-balancing

Figure 6.

Chapter 4 Storage Overview

This chapter presents the following topics:

OpenShift Container Platform storage **35**

OpenShift Data Foundation **38**

Dell storage options **40**

CSI external storage **41**

Data protection **43**

OpenShift Container Platform storage

Introduction

Stateful applications create a demand for persistent storage. All storage within OpenShift Container Platform 4.6 is managed separately from compute resources and from all networking and connectivity infrastructure facilities. The CSI API is designed to abstract storage use and enable storage portability.

This solution applies the following Kubernetes storage concepts:

- **Persistent volume (PV):** The physical LUN or file share on the storage array. PVs are internal objects against which persistent volume claims are created. PVs are unrelated to pods and pod storage life cycles.
- **PVC:** An entitlement that the user creates for the specific PV.
- **Storage class:** A logical construct defining storage allocation for a given group of users.
- **CSI driver:** The software that orchestrates persistent volume provisioning and deprovisioning on the storage array.

These resources are logical constructs that the Kubernetes container infrastructure uses to maintain storage for all the container ecosystem components that depend on storage. Developers and operators can deploy applications and provision or deprovision persistent storage without having any specific technical knowledge of the underlying storage technology.

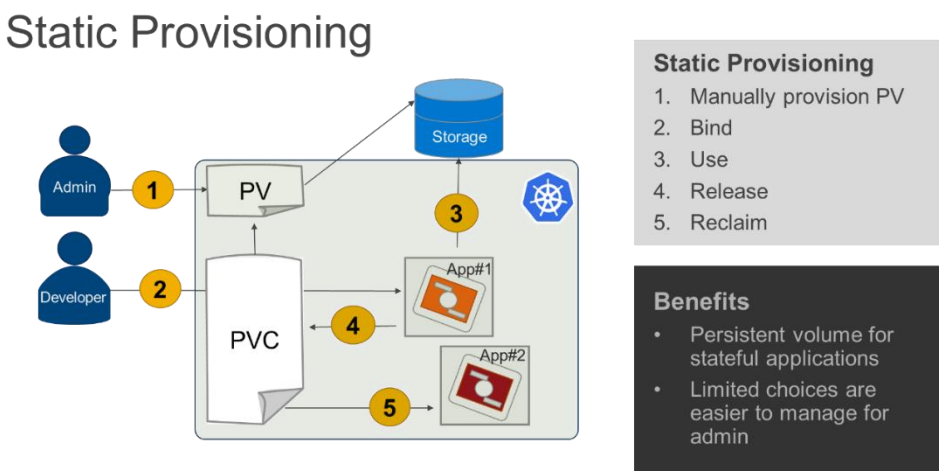
The OpenShift Container Platform administrator is responsible for provisioning storage classes and making them available to the cluster's tenants.

Storage using PVCs is consumed or used in two ways: statically or dynamically. Static storage can be attached to one or more pods by static assignment of a PV to a PVC and then to a specific pod or pods.

Static storage provisioning

With static persistent storage provisioning, an administrator preprovisions PVs for Kubernetes tenants. When a user makes a persistent storage request by creating a PVC, Kubernetes finds the closest matching available PV. Static provisioning is not the most efficient method for using storage, but it might be preferred when it is necessary to restrict users from PV provisioning.

The following figure shows the static storage provisioning workflow in this solution:



Static storage provisioning workflow

Dynamic persistent storage provisioning

Figure 7.

Dynamic persistent storage provisioning, the most flexible provisioning method, enables Kubernetes users to secure PV provisioning on demand. Dynamic provisioning has fully automated LUN export provisioning.

The following figure shows the dynamic storage provisioning workflow in this solution:

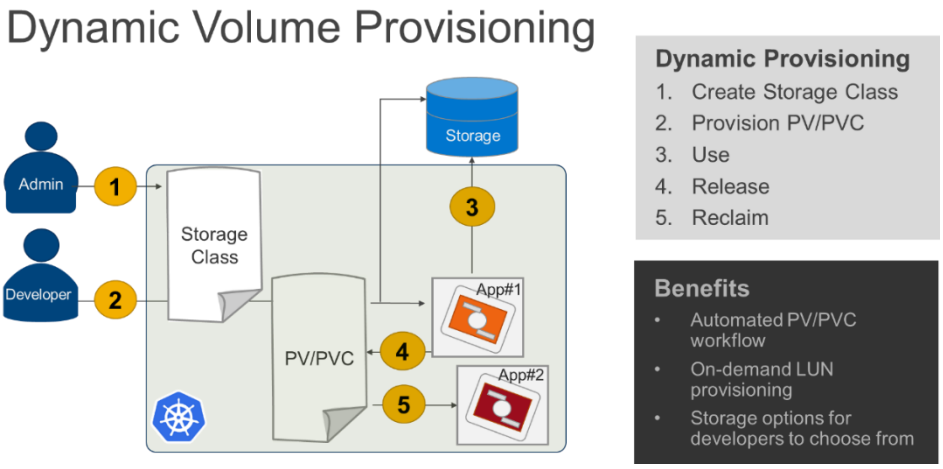


Figure 8.

Dynamic storage provisioning workflow and benefits

After a PV is bound to a PVC, that PV cannot be bound to another PVC. This restriction binds the PV to a single namespace, that of the binding project. A PV that has been created for dynamic use is a storage class object that functions as, and is automatically consumed as, a cluster resource.

PV types

OpenShift Container Platform natively supports the following PV types:

- AWS Elastic Block Store (EBS)
- Azure Disk
- Azure File
- Cinder
- Fibre Channel (FC)—can only be assigned and attached to a node
- GCE Persistent Disk
- HostPath (local disk)
- iSCSI (generic)
- Local volume
- NFS (generic)
- Red Hat OpenShift Data Foundation
- VMware vSphere

The CSI API extends the storage types that can be used within an OpenShift Container Platform solution.

PV capacity

Each PV has a predetermined storage capacity that is set in its `capacity` parameter. The storage capacity can be set or requested by a pod that is launched within the container platform. Expect the choice of control parameters to expand as the CSI API is extended and as it matures.

PV access modes

A resource provider can determine how the PV is created and can set the storage control parameters. Access mode support is specific to the type of storage volume that is provisioned as a PV. Provider capabilities determine the PV's access modes, while the capabilities of each PV determine the modes which that volume supports. For example, NFS can support multiple read/write clients, but a specific NFS PV might be configured as read-only.

Pod claims are matched to volumes with compatible access modes based on two matching criteria: access modes and size. A pod claim's access modes represent a request.

Static persistent storage

The use of generic NFS or generic iSCSI is functional and stable. However, NFS and iSCSI do not contain a mechanism to provide service continuity if access to the storage subsystem fails. Generic NFS and iSCSI do not provide the advanced storage protection support that is available using CSI drivers. The Dell Technologies Storage Engineering team validated the functionality and capability of suitable storage drivers, as described in the following table:

Table 7. Generic storage capabilities

Storage type	ReadWriteOnce	ReadOnlyMany	ReadWriteMany
HostPath (local disk)	Yes	N/A	N/A
iSCSI (generic)	Yes	Yes	N/A
NFS (generic)	Yes	Yes	Yes

OpenShift Data Foundation

Introduction

Red Hat OpenShift Data Foundation delivers a persistent data foundation, which is integrated with and optimized for OpenShift Container Platform. OpenShift Data Foundation provides file, block, and object storage classes that enable workloads for data at rest, such as databases and warehouses; data in motion, automated data pipelines; and data in action. OpenShift Data Foundation also provides services for continuous deployment models, analytics, and artificial intelligence and machine learning (AI/ML). As a part of Red Hat Data Services offerings, OpenShift Data Foundation provides a consistent experience, irrespective of the underlying infrastructure. Deployed, consumed, and managed through the OpenShift administrator console, the platform is built on Red Hat Ceph storage, offering tightly integrated, persistent data services for OpenShift and hybrid and multicloud environments.

Why OpenShift Data Foundation?

Cloud providers and system providers can offer storage for diverse workloads. Often, these storage layers are delivered using different storage interface technologies depending on the storage protocol that is addressed by an application. These storage solutions lack a service-level interface that delivers a consistent experience to users regardless of the underlying storage technologies. OpenShift Data Foundation provides data services for applications in Kubernetes that support multiple workload types across multiple cloud platforms, providing:

- **Accelerated application development:** Developer productivity depends on agile continuous integration and continuous delivery (CI/CD) pipelines and responsive infrastructure. With comprehensive support for Kubernetes, OpenShift Data Foundation automates storage provisioning alongside the provisioning of application resources, all of which are available from the OpenShift administrator console.
- **Deterministic database performance:** As databases have moved to container-based environments, the amount of stored data has grown, creating an urgent need for high-performing container-based storage. OpenShift Data Foundation provides persistent block storage for databases and supports database availability needs while providing consistency across multiple cloud platforms.
- **Simplified storage for data analytics:** Methods for analyzing data are evolving, with implications for both static data analysis and dynamic AI/ML environments. OpenShift Data Foundation lets data scientists and those who support them deploy and manage cloud-portable storage on demand, without requiring details about how data is stored or how to move datasets to other platforms.

- **Data resilience for Kubernetes:** OpenShift Data Foundation delivers container-aware backup capabilities using open standards for both persistent volume-level backup for applications and cluster protection at the namespace level. Because data protection services work with existing solutions, organizations can easily extend their backup software to container-based environments without having to invest in new methodologies and knowledge.

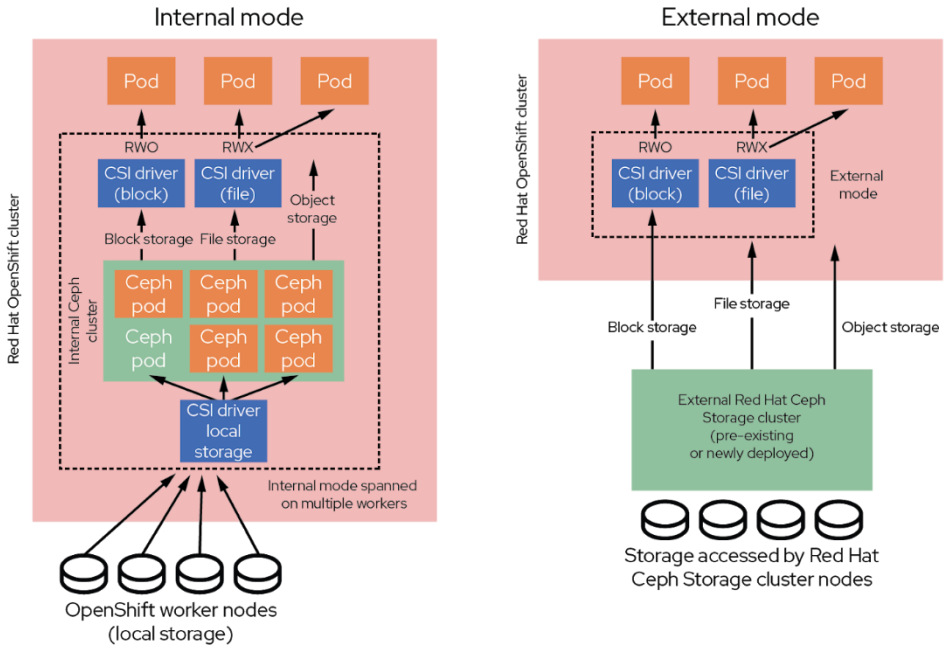
OpenShift Data Foundation architecture

Earlier releases of OpenShift Data Foundation were focused on a fully containerized Ceph cluster that is run with an OpenShift Container Platform cluster, optimized as necessary to provide block, file, or object storage with standard 3x replication. While this approach made it easy to deploy a fully integrated Ceph cluster within an OpenShift environment, it presented the following limitations:

- External Ceph clusters could not be accessed, potentially leading to a requirement for redundant storage.
- A single OpenShift Data Foundation cluster was deployed per OpenShift Container Platform instance, so the storage layer could not be mutualized across multiple clusters.
- Internal mode storage had to respect limits of total capacity and number of drives, keeping organizations from exploiting Ceph's petabyte-scale capacity.

OpenShift Data Foundation external mode overcomes these issues by allowing OpenShift Container Platform to access a separate and independent Ceph Storage cluster, as shown in the following figure.

Figure 9.



OpenShift Data Foundation accessing an external Ceph Storage cluster

Alongside traditional internal-mode storage, this option gives solution architects multiple deployment options to address their specific workload needs. These options preserve a common, consistent storage services interface to applications and workloads while providing distinct benefits:

- **Internal mode** schedules applications and OpenShift Data Foundation pods on the same OpenShift cluster, offering simplified management, deployment, speed, and agility. OpenShift Data Foundation pods can either be converged onto the same node or disaggregated on different nodes, enabling organizations to balance OpenShift compute and storage resources as they like.
- **External mode** decouples storage from OpenShift clusters, allowing multiple OpenShift clusters to consume storage from a single, external Ceph storage cluster that can be scaled and optimized as needed.

OpenShift Data Foundation support with Dell Technologies products

The PowerEdge R740xd server is optimized for storage and provides an ideal combination of performance and flexibility for software-defined storage applications. The server can accommodate various storage device configurations including 2.5-inch or 3.5-inch hard disk drives (HDDs), solid state drives (SSDs), and nonvolatile memory express (NVMe) devices.

Red Hat and Intel® have collaborated to test several configurations to optimize workloads for OpenShift Data Foundation. For information about these predefined recommended configurations optimized for PowerEdge R740xd servers, see Appendix A.

Dell storage options

For operations that are supported on Dell storage using Dell CSI drivers, see CSI Storage Feature Support.

PowerMax 2000 storage

The PowerMax 2000 storage array offers an extremely small footprint with fast end-to-end NVMe and SCM performance and rich data services, all at an attractive entry point into the PowerMax family. The array delivers the performance that your applications demand while consolidating block and file storage through iSCSI and Fibre Channel interfaces.

PowerScale storage

PowerScale is a modern, scale-out NAS solution for file storage, designed to be flexible and reliable at any scale. Regardless of the kind of data, where the data lives, or how big it gets, your data lake remains simple to manage, simple to grow, simple to protect, and simple enough to handle the most demanding workloads.

PowerStore 5000T storage

The PowerStore 5000T standard deployment model provides organizations with all the benefits of a unified storage platform for block, file, and NFS storage, while also enabling flexible growth with the intelligent scale-up and scale-out capability of appliance clusters. Automated management of resources across the cluster results in superior storage utilization and simplified administration. PowerStore's data-centric design with NVMe and hardware-enabled advanced data reduction and 4:1 DRR deliver critical performance and storage efficiency for both traditional and modern applications.

Unity XT 380F storage

The Unity XT 380F all-flash array is the entry point to the Unity XT all-flash storage series. The array provides NFS, block, and file storage for OpenShift Container Platform 4.6. Built for multi-cloud deployment, the Unity XT 380F array provides simple and affordable unified storage that is designed for high performance and low latency with the ability to simultaneously run mixed-application workloads, process inline data reduction, and provide data services without any performance impact.

Further reading

For more information about these storage options, see the following documentation:

- Dell EMC PowerMax NVMe Storage
- Dell EMC PowerScale
- Dell EMC PowerStore Scalable All Flash Storage
- Dell EMC Unity XT Unified Storage

While Dell PowerFlex was not validated as part of this project, PowerFlex storage is available for deployment with OpenShift 4.6. For separate installation and configuration steps, see the Dell EMC PowerFlex for OpenShift Installation and Configuration Guide.

CSI external storage

Introduction

OpenShift Container Platform 4.2 introduced support for the CSI operator-framework-driven API. This CSI API runs on the control-plane nodes to orchestrate and manage configuration and tear-down of data-path storage operations. While storage driver plug-in support was available in earlier Kubernetes releases, it required the integration of volume plug-ins into the core Kubernetes codebase. Kubernetes version 1.19 is integrated into OpenShift Container Platform 4.6.

Why CSI?

CSI was released in Kubernetes v1.13, replacing the volume plug-in system. Volume plug-ins were built "in-tree" (that is, as part of the Kubernetes source code), meaning that changes to various volume plug-ins provided by storage vendors had to be made in lockstep with the core Kubernetes release schedule. The CSI specification aims to standardize the exposure of block and file storage systems to workloads running on container orchestration systems such as Kubernetes. Kubernetes can now be readily extended to support any storage solution with CSI drivers that the vendor provides.

Vendors can manage the life cycle of their drivers directly, using an operator, without waiting until the next core Kubernetes release.

CSI architecture

Drivers are typically shipped as container images. These images are not platform-aware, therefore additional components are required to enable interaction between OpenShift Container Platform and the driver image. An external CSI controller running on infrastructure nodes has three containers: an attacher, provisioner, and driver. The attacher and provisioner containers serve as translators, mapping OpenShift Container Platform calls to corresponding calls to the CSI driver. No other communication to the CSI driver is allowed. On each compute node, a CSI driver daemon set is created containing the CSI driver and a CSI registrar. The registrar registers the driver with the `openshift-node` service, which then directly connects to the driver.

The following figure shows this architecture:

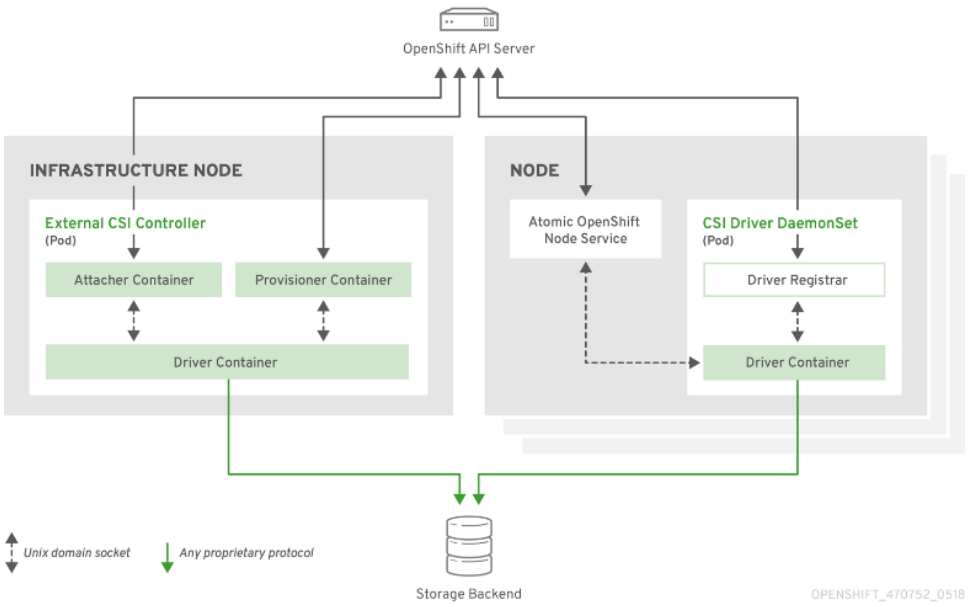


Figure 10. CSI architecture

CSI volume snapshots

Support for snapshots of CSI volumes was added in Kubernetes v1.19 and is available in OpenShift Container Platform 4.6 as a technical preview feature. OpenShift provides the CSI Snapshot Controller Operator, which manages snapshot objects. An external snapshot sidecar container must be implemented in the CSI driver to enable snapshot functionality. All Dell Technologies CSI storage drivers support snapshots. For information about production-grade cluster backup and restore functionality, see Data protection.

Storage feature support

The CSI drivers that are provided as open source by Dell Technologies enable a wide range of storage options to be employed with OpenShift. For information about all the available CSI drivers, including their features and capabilities for each of the supported Dell storage options, see the Dell Technologies CSM/CSI Drivers GitHub.

For more information about deployment and operations using CSI drivers, see the *Red Hat OpenShift Container Platform 4.6 on Dell Infrastructure Implementation Guide*.

Data protection

Introduction

As business-critical applications are developed and deployed on the OpenShift platform, organizations need to protect these applications and the application data. Protecting an OpenShift environment is not as simple as applying a traditional data backup and recovery solution to the container space. Protection for cloud-native workloads is another major challenge in container adoption.

Dell PowerProtect Data Manager

As an innovative leader in protecting Kubernetes, Dell Technologies has evolved, innovated, and integrated with OpenShift to address the needs of the new container infrastructure. Dell Technologies offers proper protection for OpenShift, including the unique complexities that come with it. Dell PowerProtect Data Manager offers a centralized platform to protect OpenShift workloads. PowerProtect Data Manager ensures HA and consistent and reliable backup and restore of workloads.

PowerProtect Data Manager protects OpenShift Kubernetes workloads, ensuring that data is easy to back up and restore and remains available, consistent, and durable in a Kubernetes workload or disaster recovery situation. PowerProtect Data Manager provides a centralized management UI where protection policies can be defined to manage the clusters, namespaces, and other OpenShift components.

PowerProtect Data Manager provides software-defined data protection, automated discovery, deduplication, operational agility, self-service, and IT governance for physical, virtual, and cloud environments, enabling you to:

- Orchestrate protection directly through an intuitive interface or empower data owners to perform self-service backup and restore operations from their native applications
- Ensure compliance and meet even the strictest service level objectives
- Leverage your existing PowerProtect appliances

For general information about PowerProtect Data Manager, see Dell EMC PowerProtect Data Manager.

For detailed information including deployment instructions for PowerProtect Data Manager, see *PowerProtect Data Manager: Protecting OpenShift Workloads*

Chapter 5 Cluster Hardware Design

This chapter presents the following topics:

Introduction 45

Cluster scaling 45

Cluster hardware planning 45

Validated hardware configuration options 47

Introduction

This chapter describes node design options that enable you to build a cluster for a wide range of workload-handling capabilities, expanding on information in the Technology and Deployment Process Overview. Usually, the platform design process ensures that the OpenShift Container Platform 4.6 cluster can meet initial workloads. The cluster must also be capable of being scaled out as the demand for workload handling grows. With a clear understanding of your workloads, it is easier to approach CPU sizing, memory configuration, network bandwidth capacity specification, and storage needs. Many operational factors can affect how the complexity of a container ecosystem affects operational latencies. It is a good practice to add a safety margin to all physical resource estimates. Our goal in providing this information is to help you get Day-2 operations under way as smoothly as possible.

Cluster scaling

The design and architecture of OpenShift Container Platform place resource hosting limits on an OpenShift cluster. Red Hat offers support for OpenShift Container Platform 4.6 up to these limits, as described in Planning your environment according to object maximums.

Control-plane node requirements

Our minimum recommended control-plane node configuration is a PowerEdge R650 server with dual Intel® Gold 6330 CPUs and 192 GB RAM. As the Red Hat resource requirements show, this node is large enough for a 250-node cluster and higher. Dell Technologies recommends that you do not scale beyond 200 nodes, which means that the proposed reference design is sufficient for nearly all deployments. The following table shows our node sizing recommendations:

Table 8. Control-plane node sizing guide

Number of compute nodes	CPU cores*	Memory (GB)
25	4	16
100	8	32
200	16	64

*Does not include provisioning of at least four cores per node for infrastructure I/O handling.

Cluster hardware planning

Design limits

The DTVD for OpenShift Container Platform 4.6 requires a minimum of four servers for a three-node cluster, with each node running as both a control-plane node and a compute node. A three-node cluster can be expanded to a standard cluster if required. You can also expand the standard cluster with more compute nodes at any time. The maximum configuration that the customized Dell Technologies deployment tools support is 210 servers.

Server, switch, and rack configuration

This guide uses a server-node base configuration for the PowerEdge R640, R650, R740xd, and R750 server nodes that can be used in each node role. For compute nodes that require add-in devices such as GPUs, we strongly recommend using PowerEdge R740xd or R750 servers. Appendix A shows the PowerEdge server baseline configurations that we used for the cluster design for our validation. The following table shows the hardware configuration of the reference design:

Table 9. Cluster configuration: Number of servers

Node name	Quantity	Configuration
CSAH	2	PowerEdge R640/R650 server configuration
Controller	3	PowerEdge R640/R650 server configuration
Compute	2 or more*	PowerEdge R640/R650 or R740xd/R750 server configuration

*A three-node cluster does not require any compute nodes. However, to expand a three-node cluster with additional compute machines, you must first expand the cluster to a five-node cluster with two additional compute nodes.

The following table provides additional cluster configuration information:

Table 10. Cluster configuration reference information

Quantity*	Description	Dell Technologies reference
1	Rack enclosure: APC AR3300 NetShelter SZ 42U	APC AR3300 NetShelter SX 42U Enclosure
1	Management switch: Dell Networking S3048-ON	Dell EMC PowerSwitch S series 1 GbE switches
2	Data switch: Dell Networking S5248F-ON or Dell Networking S5232-ON	Dell EMC PowerSwitch S series 25/40/50/100 GbE switches
7-210	CSAH, Control-plane: Dell PowerEdge R640/R650 Compute nodes: Dell PowerEdge R640/R650 or Dell PowerEdge R740xd/R750 or Dell PowerEdge XE2420	PowerEdge R640 Rack Server or PowerEdge R740xd Rack Server or PowerEdge R650 Rack Server or PowerEdge R750 Rack Server
2-4	Power distribution unit: APC metered rack PDU 17.2 kW	APC Metered Rack PDU AP8867 - 0U - 208V 3Phase IEC 309 60A 3P+PE Input / (30) C13 Output

*Rack enclosures and power distribution units are site-specific. Review the physical dimensions and power requirements in a site survey.

Validated hardware configuration options

Introduction

For validation test work in our laboratories, we used various server configurations for the DTVD for OpenShift Container Platform 4.6. Dell Technologies recommends selecting server configurations that are known to provide a satisfactory deployment experience and to meet or exceed Day-2 operating experience expectations. This chapter provides guidelines for Intel® microprocessor selection, memory configuration, local (on-server) disk storage, and network configuration.

Selecting the server processors

The Intel® Xeon Gold processor family provides performance, advanced reliability, and hardware-enhanced security for demanding compute, network, and storage workloads.

For clusters of 30 or more nodes, Dell Technologies recommends either Intel® Gold 6240 or Intel® Gold 6238 processors for Dell PowerEdge R640 and R740xd servers. Intel® Gold 6330 processors are recommended for Dell PowerEdge R650 and R750 servers.

While many sites prefer to use a single-server configuration for all node types, that option is not always cost-effective or practical.

When selecting a processor, consider the following criteria:

- **Processor core count**: The processor core count must be sufficient to ensure satisfactory performance of the workload operations and base services that are running on each node.
- **Thermal design power (TDP)**: The CPU must be suitable for the amount of heat that is removed from the server through the heat sinks and cooling airflow.
- **Ability to dissipate heat:** During validation work with high-core-count and high-TDP processors, the thermal delta (air discharge temperature minus air intake temperature) across a server was recorded at 65°F. Excessive air discharge (egress) temperature from the server can lead to a premature server-component or system failure.
- **Compute node configurations**: The design of compute nodes for use as part of your OpenShift Container Platform cluster can use many compute node configurations. Compute nodes can use Intel® or AMD-based CPU platforms. The processor architecture and core count per node selection can significantly affect the acquisition and operating cost of the cluster that is required to run your organization's application workload.

When ordering and configuring your PowerEdge servers, see the relevant guide:

- Dell EMC PowerEdge R640 Technical Guide
- Dell EMC PowerEdge R740 and R740xd Technical Guide
- Dell EMC PowerEdge R650 Technical Guide
- Dell EMC PowerEdge R750 Technical Guide

For CPU information, see Intel® Xeon Gold Processors.

Per-node memory configuration

The Dell Technologies engineering team designated 192 GB, 384 GB, or 768 GB RAM as the best choice of memory configuration based on memory usage, DIMM module capacity for the current cost, and likely obsolescence over a five-year server life cycle. We chose a mid-range memory configuration of 384 GB RAM to ensure that the memory for each CPU has multiples of three banks of DIMM slots that are populated to ensure maximum memory-access cycle speed. You can modify the memory configuration to meet your budgetary constraints and operating needs.

Consult OpenShift architectural guidance and consider your own observations from running your workloads on OpenShift Container Platform 4.6. For guidance about server memory population (location of DIMM modules in DIMM slots) and, in particular, using the firmware setting for Performance Optimized mode, see the following Dell Technologies Knowledge Base article: Dell EMC PowerEdge‑14G Memory Population Rules updated for certain server's configurations.

Disk drive capacities

The performance of disk drives significantly limits the performance of many aspects of OpenShift cluster deployment and operation. We validated deployment and operation of OpenShift Container Platform using magnetic storage drives (spinners), SATA SSD drives, SAS SSD drives, and NVMe SSD drives.

Our selection of all NVMe SSD drives was based on a comparison of cost per GB of capacity divided by observed performance criteria such as deployment time for the cluster and application deployment characteristics and performance. While there are no universal guidelines, users gain insight over time into the capacities that best enable them to meet their requirements. Optionally, you can deploy the cluster with only hard drive disk drives. In tests, this configuration has been shown to have few adverse performance consequences.

Network controllers and switches

When selecting the switches to include in the OpenShift Container Platform cluster infrastructure, consider the overall balance of I/O pathways within server nodes, the network switches, and the NICs for your cluster. When you choose to include high-I/O bandwidth drives as part of your platform, consider your choice of network switches and NICs so that sufficient network I/O is available to support high-speed, low-latency drives:

- **Hard drives:** These drives have lower throughput per drive. You can use 10 GbE for this configuration.
- **SATA/SAS SSD drives:** These drives have high I/O capability. SATA SSD drives operate at approximately four times the I/O level of a spinning hard drive. SAS SSDs operate at up to 10 times the I/O level of a spinning hard drive. With SSD drives, configure your servers with 25 GbE.
- **NVMe SSD drives:** These drives have high I/O capability, up to three times the I/O rate of SAS SSDs. We populated each node with 4 x 25 GbE NICs, or 2x 100 GbE NICS, to provide optimal I/O bandwidth.

The following table provides information about selecting NICs to ensure adequate I/O bandwidth and to take advantage of available disk I/O bandwidth:

Table 11. NIC and Storage selection to optimize I/O bandwidth

NIC selection	Compute node storage device type
2 x 25 GbE	Spinning magnetic media (hard drive)
2 x 25 GbE or 4 x 25 GbE	SATA or SAS SSD drives
4 x 25 GbE or 2 x 100 GbE	NVMe SSD drives

True network HA fail-safe design demands that each NIC is duplicated, permitting a pair of ports to be split across two physically separated switches. A pair of PowerSwitch S5248F-ON switches provides 96 x 25 GbE ports, enough for approximately 20 servers. This switch is cost-effective for a compact cluster. While you could add another pair of S5248F-ON switches to scale the cluster to a full rack, consider using PowerSwitch S5232F-ON switches for a larger cluster.

The PowerSwitch S5232F-ON provides 32 x 100 GbE ports. When used with a four-way QSFP28 to SFP28, a pair of these switches provides up to 256 x 25 GbE endpoints, more than enough for a rackful of servers in the cluster before more complex network topologies are required.

Low latency in an NFV environment

NFV-centric data centers require low latency in all aspects of container ecosystem design for application deployment. This requirement means that you must give attention to selecting low-latency components throughout the OpenShift cluster. We strongly recommend using only NVMe drives, NFV-centric versions of Intel® CPUs, and, at a minimum, the PowerSwitch S5232F-ON switch. For specific guidance, consult the Dell Technologies Service Provider support team.

Power configuration

Dell Technologies strongly recommends that all servers are equipped with redundant power supplies and that power cabling provides redundant power to the servers. Configure each rack with pairs of power distribution units (PDUs). For consistency, connect all right-most power supply units (PSUs) to a right-side PDU and all left-most PSUs to a left-side PDU. Use as many PDUs as you need, in pairs. Each PDU must have an independent connection to the data center power bus.

The following figure shows an example of the power configuration that is designed to ensure a redundant power supply for each cluster device:

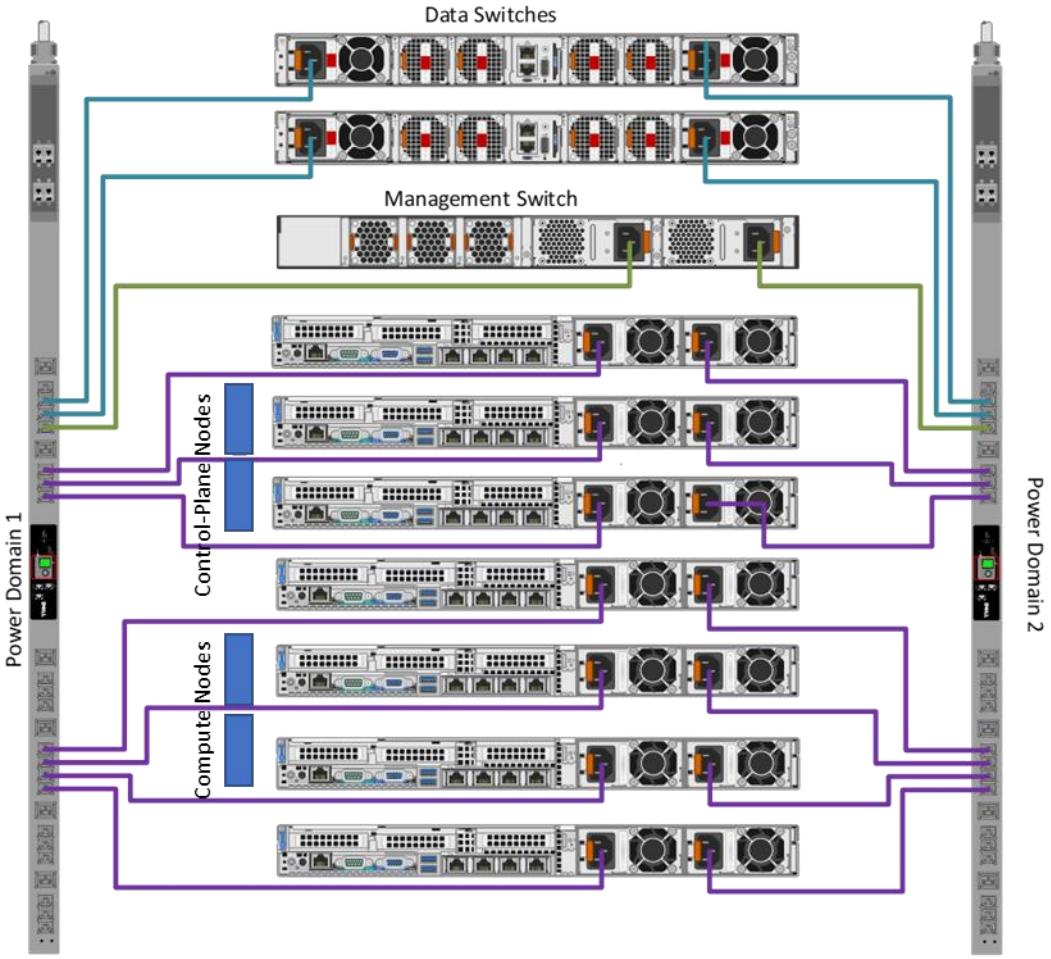


Figure 11. **PSU to PDU power template**

Chapter 6 Use Cases

This chapter presents the following topics:

Overview **52**

Enterprise applications **52**

Networking **55**

Data analytics and artificial intelligence **57**

Overview

This chapter describes how an OpenShift Container Platform 4.6 solution supports several different uses cases across both enterprise and service provider markets. The examples in this chapter include enterprise application development and deployment, telecommunications service provider operations, and data analytics and artificial intelligence.

Enterprise applications

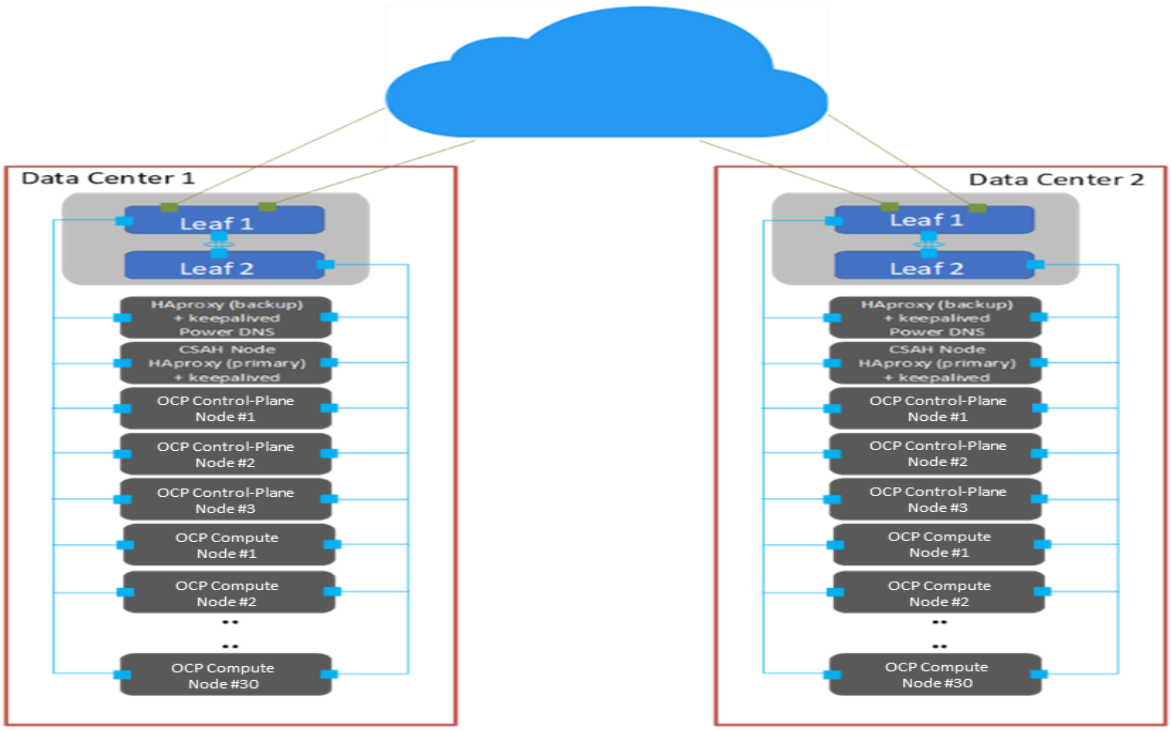
Today's applications are developed using cloud-native principles. These applications offer the agility, speed, and composability of microservices architecture. Enterprise deployment of these applications comes with the following additional requirements:

- Enterprises require geographically distributed deployment of container platforms as a means of *disaster avoidance*.
- Large enterprise deployments might cater to many internal and external partners and customers. This consideration requires multitenancy and user access roles in the container platform deployments. Security and isolation are required between microservices.
- Infrastructure requirements to meet the application needs of network connectivity, storage capacity, and compute.

Site deployment models

Enterprises must deploy a single cluster or multiple highly scalable large clusters in each data center. Dell Technologies strongly discourages spanning a single cluster across multiple geographical sites because Kubernetes has low latency requirements. One of the key decisions we made regarding multisite OpenShift Container Platform 4.6 deployment was to deploy multiple OpenShift clusters across multiple sites. A major advantage of this model of deployment is disaster avoidance. Service can continue even when a disaster occurs at a site.

The following figure shows a multisite deployment of this solution:



Multisite OpenShift Container Platform deployment

Figure 12.

As the diagram shows, key technical components of the solution include:

- OpenShift Container Platform 4.6
- Load balancer: Global traffic manager (GTM) and local traffic manager (LTM)
- Data center hardware infrastructure

Role-based access control

Access control has implications for what multitenancy means throughout the infrastructure. Portal access and views, logging information, and usage information must be linked to the user role. For example:

- A provider administrator must be able to see usage and metering information for the entire infrastructure.
- A tenant administrator requires access only to the infrastructure that is assigned to that tenant.
- Tenant users require access only to assets and resources that they are permitted to manage.

Role-based access control (RBAC) in OpenShift Container Platform 4.6 can be linked to your Microsoft Active Directory identity management environment or other supported identity managers. This link gives control over user and group access to the container ecosystem infrastructure and services, providing a good foundation for multitenancy support. The following table shows the roles that OpenShift Container Platform 4.6 supports:

Table 12. Role-based access control in OpenShift Container Platform 4.6

Role	Description
admin	Project manager
basic-user	User who can get information about projects and users
cluster-admin	A superuser who can perform any action in any project
cluster-status	User who can get cluster status information
edit	User who can modify objects in a project
self-provisioner	User who can create their own projects
view	User who can see most objects in a project
cluster-reader	User who can read, but not view, objects in the cluster

Security and isolation

OpenShift Container Platform 4.6 is built on the concept that each project that is running within a cluster can be isolated from every other project. The project manager must have the administrative privilege to be able to see any other project in the cluster.

Performance monitoring and logging

Cloud service providers typically require the ability to monitor and report on system utilization. OpenShift Container Platform 4.6 includes Prometheus system monitoring and metering and provides the capability for extensive data logging. For more information about obtaining cluster resource consumption to drive usage billing through third-party application software, see the following Red Hat documentation:

- About cluster monitoring
- Examples of using metering
- About cluster logging and OpenShift Container Platform

The cluster monitoring operator controls the monitoring components that are deployed, and the Prometheus operator controls Prometheus and Alert manager instances. The platform monitors the following stack components:

- Stack component
- CoreDNS
- Elasticsearch
- Etcd
- Fluentd
- HAProxy
- Image registry
- Operator Lifecycle Manager (OLM)
- Telemeter client
- Kubelets
- Kubernetes apiserver
- Kubernetes controller manager

- Kubernetes scheduler
- Metering
- OpenShift apiserver
- OpenShift controller manager

Networking

Introduction

A typical telecommunications (telecom) company sells telecom-oriented applications as a service to its consumers. Telecom use-case requirements vary depending on the virtual network functions (VNFs) that are being serviced. These functions include:

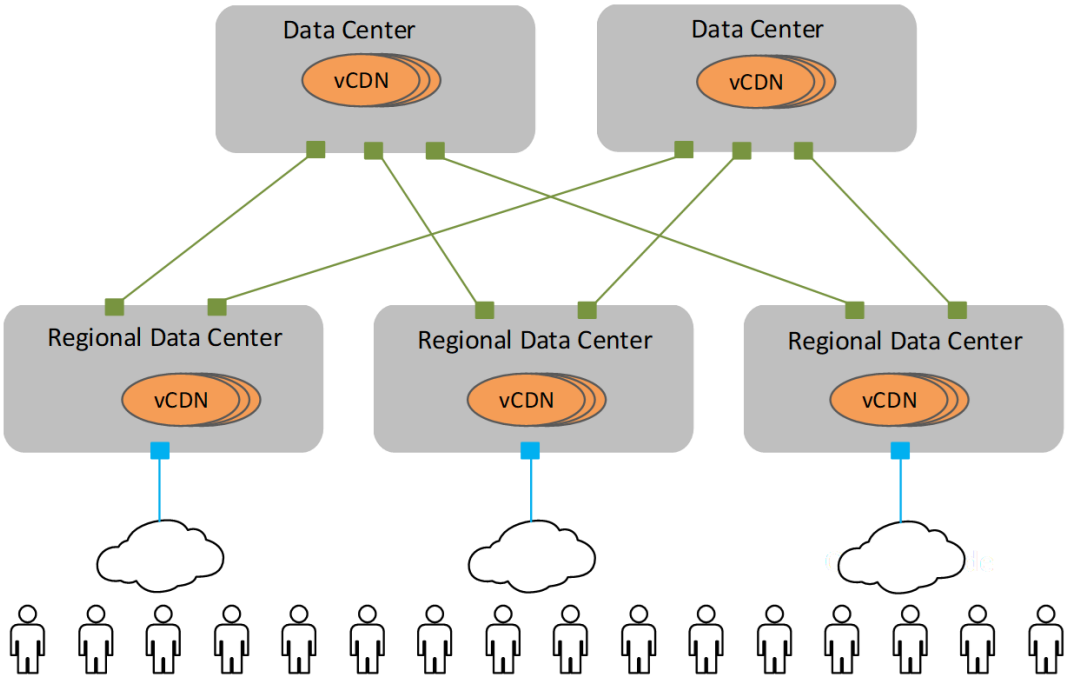
- Content delivery network (CDN)
- Edge infrastructure and towers of power
- NFV management and operations (NFV-MANO)
- Software-defined networking (SDN) and SD-WAN management
- Radio access networks (RAN) and 5G, and their component service infrastructures
- Multiaccess Edge Computing (MEC)
- Core network and 5G Next Generation Core (NGC)

This use case identifies some key design factors for a telco container platform.

Content delivery network

Online video consumption has grown in recent years. High-quality video delivery over public networks requires a CDN. To handle growth, many operators are considering the virtualization of the CDN, giving them the ability to scale CDN on demand. CDN virtualization permits simple provisioning and sharing of resources with other telecom services, simplifying operations and avoiding costly dedicated infrastructure.

The following figure shows a virtual CDN (vCDN):



Virtual content delivery network (vCDN)

Figure 13.

vCDN platform requirements

A vCDN stack requires the following principal capabilities:

- Large application storage space to store video and other files
- High-speed and low-latency network options to serve the content
- Rapid ramp-up of on-demand processing capacity

OpenShift Container Platform 4.6 on PowerEdge hardware platforms meets these demands by providing the following capabilities:

- CSI storage drivers for Dell Unity, PowerMax, PowerFlexOS, and PowerScale (formerly Isilon) storage systems are available and have been certified by Red Hat. These drivers can be integrated into your OpenShift Container Platform deployment using the new CSI plug-ins.
- High-speed (25 GbE/100 GbE) network interfaces of Dell Technologies server and switch portfolios meet the networking needs of network I/O-intensive applications.
- OpenShift Data Foundation (based on Ceph) is supported as part of OpenShift Container Platform 4.6.
- Multus CNI plug-in support by which additional networks can be added to each container so that the container can meet capacity needs on targeted networks.
- SR-IOV is now natively supported by OpenShift Container Platform. Red Hat provides an SR-IOV Operator over OperatorHub, enabling administrators to manage virtual functions on nodes through Kubernetes CRDs.
- DPDK is available in OpenShift Container Platform 4.6 as a Technical Preview feature. Currently, DPDK is not supported by Red Hat in production environments.

- OpenShift Container Platform now supports running workloads inside VMs that are natively managed and deployed by OpenShift Container Platform.
- For low-latency workloads, you can configure compute nodes running Red Hat CoreOS to use the real-time kernel, which provides consistent low-latency performance.
- Compute nodes can also be deployed at the edge, managed by a central cluster located in a data center.
- Small footprint clusters (three-node clusters) can also be deployed at the edge for production environments.
- Local storage on edge nodes can be provisioned for use by applications, allowing edge nodes to operate independently of an external storage array, at the cost of resiliency.
- Telco applications generally use huge pages. In OpenShift Container Platform, applications can allocate and consume huge pages.
- OpenShift Container Platform 4.6 includes support for IPv6.

Container ecosystem clusters for large deployments can span across multiple racks. We highly recommend using leaf-spine networking when scaling to more than three racks per cluster. The deployment of large clusters requires significant modification of the Ansible playbooks that were generated to facilitate large-scale deployment.

Data analytics and artificial intelligence

Introduction

Enterprises are rapidly increasing their investments in infrastructure platforms to support data analytics and AI, including the more specific AI disciplines of ML and deep learning (DL). All these disciplines benefit from running in containerized environments. The benefits of running these applications on OpenShift Container Platform are available to developers, data scientists, and IT operators.

For simplicity, we use the term “data analytics as a service” (DAaaS) for analytics and AI that are operated and instantiated in a containerized environment. OpenShift Container Platform enables operators to create a DAaaS environment as an extensible analytics platform with a private cloud-based delivery model. This delivery model makes various tools available for data analytics and can be configured to efficiently process and analyze huge quantities of heterogeneous data from shared data stores.

The data analytics life cycle, particularly the ML life cycle, is a multiphase process of integrating large volumes and varieties of data, abundant compute power, and open-source languages, libraries, and tools to build intelligent applications and predictive outcomes. At a high level, the life cycle consists of these phases:

- **Data acquisition and preparation**—Ensures that the input data is complete and of a high quality.
- **Modeling creation**—Includes training, testing, and selection of the model with the highest prediction accuracy.

- **Model deployment**—Includes inferencing in the application development and operations processes.

Key challenges

Data scientists and engineers are primarily responsible for developing modeling methods that ensure that the selected outcome continues to provide the highest prediction accuracy. The key challenges that data scientists face include:

- Selection and deployment of the right AI tools (such as Apache Spark, TensorFlow, and PyTorch)
- Complexities and time required to train, test, select, and retrain the AI model that provides the highest prediction accuracy
- Slow execution of AI modeling and inferencing tasks because of a lack of hardware acceleration
- Limited IT operations to provision and manage infrastructure
- Collaboration with data engineers and software developers to ensure input data hygiene and successful AI model deployment in application development processes

Containers and Kubernetes are key to accelerating the data analytics life cycle because they provide data scientists and IT operators with the agility, flexibility, portability, and scalability that is needed to train, test, and deploy ML models.

OpenShift Container Platform provides all these benefits. Through its DevOps capabilities and integration with hardware accelerators, the platform enables better collaboration between data scientists and software developers. OpenShift Container Platform also accelerates the roll-out of analytics applications to departments as needed. The benefits include the ability to:

- Empower data scientists with a consistent, self-service-based, cloud-like experience which includes:
  - Giving data scientists the flexibility and portability to use containerized ML tools of their choice to quickly build, scale, reproduce, and share ML modeling results in a consistent way with peers and software developers.
  - Eliminating dependency on IT to provision infrastructure for iterative, compute-intensive ML modeling tasks.
- Accelerate compute-intensive ML modeling and inferencing jobs:

  On-demand access to high-performance hardware can seamlessly meet the high compute resource requirements to help determine the best ML model, providing the highest prediction accuracy.
- Streamline the development and operations of intelligent applications:

  Extending OpenShift DevOps automation capabilities to the ML life cycle enables collaboration between data scientists, software developers, and IT operations so that ML models can be quickly integrated into the development of intelligent applications.

MLPerf on OpenShift

A recent white paper explored the implications of running resource-intensive ML applications on top of OpenShift. MLPerf benchmarks are an independent valuation of performance for various parts of the machine learning ecosystem, including both the cloud

and hardware platforms being used. The MLPerf training and inference benchmarks were run on top of OpenShift and compared to NVIDIA's MLPerf benchmark results. The NVIDIA MLPerf benchmarks were not run on top of a container automation platform. The results indicated that the addition of the OpenShift platform did not hamper the performance of intensive ML applications and demonstrated that OpenShift provides valuable benefits for running ML applications in production environments.

Kubeflow ML on OpenShift

One example of ML on OpenShift Container Platform is the work that Dell Technologies and Red Hat did to deploy Kubeflow on OpenShift.

Kubeflow is an open-source Kubernetes-native platform for ML workloads that enables service providers to accelerate their ML/DL projects. Based originally on Google's use of TensorFlow on Kubernetes, Kubeflow is a composable, scalable, portable ML stack that includes components and contributions from a variety of sources and organizations. Kubeflow bundles popular ML/DL frameworks such as TensorFlow, MXNet, PyTorch, and Katib with a single deployment binary file. By running Kubeflow on OpenShift Container Platform, you can quickly operationalize a robust ML pipeline.

For more information, see the Machine Learning Using Red Hat OpenShift Container Platform (this white paper is based on the OpenShift Container Platform 4.2 release).

For more information, see Kubeflow: The Machine Learning Toolkit for Kubernetes.

Spark analytics on OpenShift

An example of large-scale data analytics being run on OpenShift Container Platform is the Dell Spark on Kubernetes Solution for Data Analytics. Apache Spark, a unified analytics engine for big data and ML, is one of the largest open-source projects in data processing. Data scientists want to run Spark processes that are distributed across multiple systems to have access to additional memory and computing cores. OpenShift orchestrates the creation, placement, and life cycle management of those Spark processes across a cluster of servers by using container virtualization to host the processes.

For more information, see the Spark on Kubernetes reference architecture guide.

SQL Server big data clusters on OpenShift

Another example of big data analytics being run on OpenShift is the Dell Technologies solution for Microsoft SQL Server 2019 Big Data Clusters.

SQL Server Big Data Clusters enable deployment of scalable clusters consisting of SQL Server, Spark, and HDFS containers running on Kubernetes. These components run side by side to enable you to read, write, and process big data so that you can easily combine and analyze your high-value relational data with high-volume big data. OpenShift Container Platform is one of the Kubernetes platforms on which you can run SQL Server Big Data Clusters.

For more information, see the Microsoft SQL Server 2019 Big Data Clusters White Paper on the Dell Technologies Info Hub for SQL Server.

Chapter 7 References

This chapter presents the following topics:

Dell Technologies documentation 61

Red Hat documentation 61

Other resources 62

Dell Technologies documentation

The following Dell Technologies documentation provides additional information. Access to these documents may depend on your login credentials. If you do not have access to a document, contact your Dell Technologies representative.

- *Red Hat OpenShift Container Platform 4.6 on Dell EMC Infrastructure Implementation Guide*
- *Dell EMC PowerEdge R640 Technical Guide*
- *Dell EMC PowerEdge R740 and R740xd Technical Guide*
- *Dell EMC PowerEdge R650 Technical Guide*
- *Dell EMC PowerEdge R750 Technical Guide*
- *Machine Learning Using the Dell EMC Ready Architecture for Red Hat OpenShift Container Platform (this document is based on OpenShift Container Platform 4.2).*
- *Dell EMC Unity: Best Practices Guide*
- *Dell Technologies Solutions Info Hub for Communication Service Providers*
- *Dell EMC Ready Stack Converged Infrastructure*
- *Dell EMC PowerProtect Data Manager Protecting OpenShift Workloads*
- *Dell EMC PowerMax Storage*
- *Dell EMC PowerScale Storage*
- *Dell EMC PowerStore Storage*
- *Dell EMC Unity XT Storage*
- *Dell EMC PowerFlex for OpenShift Installation and Configuration Guide*
- *Dell Technologies Container Storage Modules and CSI Drivers GitHub*

Note: For links to additional relevant documentation, see the Red Hat OpenShift Container Platform section of the Dell Technologies Info Hub for Containers.

Red Hat documentation

The following Red Hat documents provide additional information:

- *Installing on Bare Metal*
- *What are Operators?*
- *Understanding Red Hat OpenShift Service Mesh*
- *Recommended Cluster Scaling Practices*
- *Understanding the monitoring stack*
- *Examples of using metering*
- *Understanding cluster logging*
- *Machine Management*

- *Planning your environment according to object maximums*

Other resources

The following resources provide additional information:

- Intel® Xeon Gold Processors
- Kubeflow: The Machine Learning Toolkit for Kubernetes
- Prometheus: From metrics to insight
- Operating etcd clusters for Kubernetes
- NVIDIA Tesla GPU Documentation

Appendix A Hardware Configuration

This appendix presents the following topics:

Overview 64

Dell PowerEdge R640 node BOM 64

Dell PowerEdge R740xd node BOM 65

Dell PowerEdge R650 BOM 66

Dell PowerEdge R750 BOM 67

Dell PowerStore 1000T BOM 67

Dell Unity 380F BOM 69

Dell PowerMax BOM 69

OpenShift Data Foundation data node configurations 70

Overview

Each table in this appendix shows the Bill of Materials (BOM), the list of recommended parts for each node. The memory, CPU, NIC, and drive configurations that are shown in the tables are preferred but not mandated.

Note: When orders are placed, the Dell Technologies ordering center adds new SKUs and substitutes those that are shown in the tables with current local SKUs.

Dell PowerEdge R640 node BOM

The following table lists the key recommended parts per node. The memory, CPU, NIC, and drive configurations in the table are preferred but not mandated.

Table 13. PowerEdge R640 baseline server BOM

Qty	SKU	Description
1	210-AKWU	PowerEdge R640 server
1	329-BEIJ	PowerEdge R640 MLK motherboard
1	321-BCQQ	2.5-inch-inch chassis with up to 10 hard drives, 8 NVMe drives, and 3 PCIe slots, 2 CPU only
2	338-BTSI	Intel Xeon Gold 6238 2.1G, 22C/44T, 10.4GT/s, 30.25M Cache, Turbo, HT (140W) DDR4-2933
1	370-ABWE	DIMM blanks for system with 2 processors
2	412-AAIQ	Standard 1U heatsink
1	370-AEVR	3200 MT/s RDIMMs
1	370-AAIP	Performance-optimized
12	370-AEVN	32 GB RDIMM, 3200MT/s, Dual Rank
1	405-AAJU	HBA330 12 Gbps SAS HBA controller (NON-RAID), minicard
1	385-BBKT	iDRAC9, Enterprise
1	379-BCQV	iDRAC Group Manager, enabled
1	379-BCSG	iDRAC, legacy password
1	379-BCRB	DHCP with Zero Touch Configuration
1	330-BBGN	Riser config 2, 3 x 16 LP
1	406-BBLG	Mellanox ConnectX-4 Lx Dual Port 25 GbE SFP 28 rNDC
1	406-BBLD	Mellanox ConnectX-4 Lx dual port 25 GbE SFP28 NIC, low profile
1	429-AAIQ	No internal optical drive
1	384-BBQI	8 performance fans for the R640 server
1	450-ADWS	Dual, hot-plug, redundant power supply (1+1), 750W
2	492-BBDH	C13 to C14, PDU Style, 12 AMP, 2 ft (0.6 m) power cable, North America

Qty	SKU	Description
1	800-BBDM	UEFI BIOS boot mode with GPT partition
1	770-BBBC	ReadyRails sliding rails without cable management arm
1	366-0193	Std BIOS setting power management—maximum performance
2 min – 8 max	400-BELT	Dell 1.6 TB, NVMe, Mixed Use Express Flash, 2.5 SFF Drive, U.2, P4610 with Carrier
2	400-AZQO	800 GB SSD SAS Mix Use 12Gbps512e 2.5in Hot-plug AG Drive, 3 DWPD, 4380 TBW
1	403-BCHI	BOSS Cntrl + 2 M.2 240G, R1, LP1

Dell PowerEdge R740xd node BOM

The following table shows the PowerEdge Server R740xd baseline configurations that are used in this design for OpenShift Container Platform 4.6.

Table 14. PowerEdge R740xd baseline server BOM

Qty	SKU	Description
1	210-AKZR	PowerEdge R740XD Server
1	329-BEIK	PowerEdge R740/R740XD MLK motherboard
1	321-BCRC	Chassis up to 24 x 2.5 -inch hard drives including 12 NVME drives, 2 CPU configuration
1	338-BTSI	Intel Xeon Gold 6238 2.1G, 22C/44T, 10.4GT/s, 30.25M Cache, Turbo, HT (140W) DDR4-2933
1	412-AAIR	Standard 2U heatsink
1	370-AEVR	3200 MT/s RDIMMs
12	370-AEVN	32 GB RDIMM, 2933MT/s, Dual Rank
1	780-BCDI	No RAID
1	405-AANK	HBA330 controller adapter, low profile
1	365-0354	CFI, standard option not selected
1	385-BBKT	iDRAC9, Enterprise
1	379-BCQV	iDRAC Group Manager, enabled
1	379-BCSG	iDRAC, legacy password
1	385-BBLG	Static IP
1	330-BBHD	Riser Config 6, 5 x 8, 3 x1 6 slots
1	406-BBLG	Mellanox ConnectX-4 Lx Dual Port 25 GbE SFP28 rNDC
1	406-BBLE	Mellanox ConnectX-4 Lx Dual Port 25 GbE SFP28 network interface controller
1	384-BBPZ	6 performance fans for R740/740XD
1	450-ADWM	Dual, hot-plug, redundant power supply (1+1), 1100W

Qty	SKU	Description
1	492-BBDH	C13 to C14, PDU Style, 12 AMP, 2 ft (0.6 m) power cable, North America
1	325-BCHU	PowerEdge 2U standard bezel
1	800-BBDM	UEFI BIOS Boot Mode with GPT partition
1	770-BBBQ	ReadyRails sliding rails without cable management arm
1	366-0193	Std Bios setting power management - maximum performance
1	403-BCHP	BOSS Cntrl + 2 M.2 240G, R1, FH
Select *one* of the following rows:		
1 to 24	Check part at time of ordering	800 GB, 1.92 TB, or 3.84 TB SSD SAS mixed use 12 Gbps 512e 2.5 - inch hot-plug AG drive with carrier, 3 DWPD, 4380 TBW, CK
1 to 12	Check part at time of ordering	Dell 1.6 TB, 3.2 TB, or 6.4 TB, NVMe, mixed use express flash, 2.5 SFF drive, U.2, P4610 with carrier, CK

Dell PowerEdge R650 BOM

The following table shows the PowerEdge Server R650 baseline configurations that are used in the Dell design for OpenShift Container Platform 4.6.

Table 15. PowerEdge Server R650 BOM

QTY	SKU	Description
1	338-BZXK	Gold 6330 2G, 42M, 205W
1	338-BZXK	Gold 6330 2G, 42M, 205W
16	370-AEVQ	16 GB RDIMM, 3200MT/s, Dual Rank
1	450-AIQZ	Dual, Hot-plug, PSU 1+1, 1400W, MM
1	528-CRVW	iDRAC9 data center 15G
1	340-CUQN	R650 Ship 4x3.5, 10x2.5
1	321-BGHG	8x2.5 Chipset NVMe RAID Config, 2CPU
1	405-AAZE	PERC H755N Front
1	330-BBRP	Riser C0-2, 3x16 LP, HL
1	403-BCMB	BOSS-S2 Cntrl + 2 M.2 480G
3	400-BLKD	1.6 TB, NVMe, 2.5 Dr, MU, P5600
1	540-BCOF	Mlnx ConX5 DP 10/25Gbe SFP28 OCP3.0
1	540-BCMQ	Mlnx ContX-5 DP 25 Gb SFP Adpt, LP

Dell PowerEdge R750 BOM

The following table shows the PowerEdge Server R750 baseline configurations that are used in this design for OpenShift Container Platform 4.6:

Table 16. PowerEdge Server R750 BOM

QTY	SKU	Description
1	338-BZXK	Gold 6330 2G, 42M, 205W
1	338-BZXK	Gold 6330 2G, 42M, 205W
16	370-AEVQ	16 GB RDIMM, 3200MT/s, Dual Rank
1	450-AIQZ	Dual, Hot-plug, PSU 1+1, 1400W, MM
1	528-CRVW	iDRAC9 Datacenter 15G
3	400-BLKD	1.6 TB, NVMe, 2.5 Dr, MU, P5600
1	321-BGET	8x2.5" NVMe RAID
1	330-BBRW	Riser Config2, Full Length,4x16,2x8 slots
1	540-BCNM	Mlnx ContX-5 DP 25Gb SFP Adpt, FH
1	403-BCMB	BOSS-S2 Cntrl + 2 M.2 480G
1	405-AAZE	PERC H755N Front
1	540-BCOF	Mlnx ConX5 DP 10/25Gbe SFP28 OCP3.0

Dell PowerStore 1000T BOM

The following table shows the PowerEdge Server 1000T baseline configurations that are used in this design for OpenShift Container Platform 4.6.

Table 17. PowerEdge 1000T BOM

Qty	SKU	Description
1	210-ASTZ	PowerStore 1000T BASE ENC. FLD INST
1	343-BBMR	BASE UNIT CONFIG KIT
1	370-AEZP	384 GB Appliance DIMM 192 GB Per Node
8	400-BGGI	P1 25 X 2.5 NVMe SED SSD 1.92 TB
1	528-BTZK	PowerStore Base SW
1	406-BBQI	10 GBE OPTICAL 4 PORT CARD PAIR
1	565-BBJP	16 GB FC 4 PORT IO MODULE PAIR

Qty	SKU	Description
1	565-BBJR	25 GBE TWINAX 4 PORT IO MODULE PAIR
1	450-AIOM	1800 WATT POWER SUPPLY PAIR
1	800-BBQV	Thank you for buying Dell
1	379-BDPD	ISG Product
1	825-9489	HW WRTY-SVC PS 1000
1	825-9490	PS NBD OS PS 1000 1YR
1	825-9496	PS TECH SPT PS 1000 1YR
1	989-3439	INFO PS TECH SPT CONTACT ENTERPRISE
8	828-4814	PS NBD SSD LOW ADD ON 1Y
1	800-BBQV	Thank you for buying Dell
1	992-6149	INFO INSTL CSTM REQUIRED
1	332-1286	US Order

Dell Unity 380F BOM

The following table shows the Dell Unity 380F baseline configurations that are used in this design for OpenShift Container Platform 4.6:

Table 18. Dell Unity 380F BOM

Qty	SKU	Description
8	D4F-2SFXL2-1920	D4F 1.92 TB ALL FLASH 25X2.5 SSD
1	D4ODPEKITAF	UNITY 380F DPE INSTALL KIT
2	C13-PWR-12	2 C13 CORDS NEMA 5-15 125 V 10A - NON DPE
1	D4BD6C25FAFLL	UNITY 380F DPE 25 X 2.5 DELL FLD RCK
1	D4SFP16FAF	UNITY CNA 4X16 GB FC SFPS AF
1	D4SL25IO4PTAF	UNITY 2X4 PORT I/O 25 GBE OPT AF
1	M-PSM-HWE-005	PROSUPPORT 4HR/MC HARDWARE SUPPORT
1	458-002-526	UNITY AFA BASE SOFTWARE=IC
1	M-PSM-SWE-005	PROSUPPORT 4HR/MC SOFTWARE SUPPORT
1	PS-PD-UXAFXDP	PD FOR UNITY XT AF

Dell PowerMax BOM

The following table shows the baseline configurations of the PowerMax (formerly VMAX) storage array that are used in this design for OpenShift Container Platform 4.6:

Table 19. Dell PowerMax BOM

Qty	SKU	Description
1	EH-SYS1-3D	POWERMAX 2000 SYS BAY1 3D
1	EH-PSNT-3D	POWERMAX 2000 SYS
1	EH-SKINS	POWERMAX 2000 SIDE PANELS
1	EH-VBX-1024	POWERMAX 2000 PRO BASE 1024 GB
8	QNDN3192051	POWERMAX 2000 RAID 5(3+1) 1.92 TB
1	EH-PCBL3DHR	PWR CBL HBL-RSTOL 3D
2	EH-ACON3P-50	ADPTR AC 3PH 50A W3-4IN CONDUIT ADPTR
1	EH-VBBASE-KIT	POWERMAX 2000 PB BASE INSTALL KIT
2	EH-DE24	POWERMAX 2000 DIRECT 24 SLT DR ENCL
1	EHX-BEDIR	POWERMAX 2000 PRO DIR
1	EH-1024BASE	POWERMAX 2000 BASE 1024 GB
1	EH-FE00800T	POWERMAX 2000 8MM 10GIGE

Qty	SKU	Description
1	EH-COMPRESS	POWERMAX 2000 HDW COMPRESSION
1	QNDN31920S1	POWERMAX 2000 1.92 TB SPARE
2	EH-1600MOD	POWERMAX 2000 FLASH MOD 1600
8	E-GE-ISCSI	VMAX VG GIGE ISCSI PORT TRACKING MODEL
3000	E-DRR	POWERMAX 2/8K DATA REDUCTION RESERVATION
1	E-Q118E	ELM TRACKING MODEL
13	E-OPROVISION	OPROVISION FACTOR TRACKING MODEL
1	WKPROFILE-BAL	VMAX VG WORKPROFILE BALANCED
1	EH-MGMT	EMBEDDED MANAGEMENT POWERMAX 2000 TRK
1	M-PSM-HW-020	PROSUPPORT 4HR/MC HARDWARE SUPPORT
11	458-002-223	POWERMAX PRO SUITE OS 1 TB=CC
1	M-PSM-SW-020	PROSUPPORT 4HR/MC SOFTWARE SUPPORT
1	450-001-644	POWERMAX PRO SUITE=IC
1	M-PSM-SW-020	PROSUPPORT 4HR/MC SOFTWARE SUPPORT
11	450-001-645	POWERMAX PRO SUITE 1 TB=CC
1	M-PSM-SW-020	PROSUPPORT 4HR/MC SOFTWARE SUPPORT
1	PS-PD-PMAX2DP	PD FOR POWERMAX 2000
1	PSINST-ESRS	ZERO DOLLAR ESRS INSTALL
1	CE-VALPAKPMAXSADM	POWERMAX/VMAX AF STORAGE ADMIN VALUEPAK

OpenShift Data Foundation data node configurations

Three workload-optimized configurations for OpenShift Data Foundation external data nodes have been developed for consumption based on testing that Red Hat and Intel conducted. For details of possible storage node configurations, see the Data nodes for Red Hat OpenShift data sheet. Each configuration is available in a Base configuration and a Plus configuration. With this range of configuration choices, you can easily pick data nodes of the right size for your particular workload, whether it be edge computing, high-capacity big data analytics, or latency-sensitive database transactions. Each configuration features the most appropriate Intel® Xeon Scalable processor, Intel® SSDs, and Intel® networking products.

In the near future, configurations will also be available with 3rd Gen Intel® Xeon Scalable processors and 2nd Gen Intel® Optane SSDs. Core-for-core, these processors offer industry-leading performance on popular databases, HPC workloads, and AI. The

configurations will take advantage of performance enhancements and increase business value across a wide range of workloads.

Using these data node configurations, you can achieve the portability, consistency, and performance that you need to run your specific workloads, with confidence that the overall platform is fully interoperable with your existing infrastructure.