HITACHI

Hitachi Vantara

Hitachi Integrated Systems Solution with Red Hat OpenShift Virtualization and Hitachi Virtual Storage Platform One Block

Reference Architecture Guide

MK-SL-374-02
November 2025

The open source content used in Hitachi Vantara products may be found within the Product documentation or you may request a copy of such information (including source code and/or modifications to the extent the license for any open source requires Hitachi make it available) by sending an email to OSS_licensing@hitachivantara.com.

Feedback

Hitachi Vantara welcomes your feedback. Please share your thoughts by sending an email message to Docs-Feedback@hitachivantara.com. To assist the routing of this message, use the paper number in the subject and the title of this white paper in the text.

Revision history

Changes	Date
Minor updates	November 2025
Add support for Migration Toolkit for Virtualization (MTV) Operator.	September 2025
Initial release	August 2025

Reference Architecture Guide

This paper presents best practices and use cases for a reference configuration of Red Hat OpenShift Container Platform (OCP), enhanced with OpenShift Virtualization and supported by Hitachi Virtual Storage Platform One (VSP One) as a robust backend storage system. It leverages the latest capabilities and services to create, manage, and store virtual machines alongside standard containerized applications. It also covers the migration of virtual machines from other source providers, such as VMware vSphere, to Red Hat OpenShift Virtualization to store the VMs on VSP One storage systems.

A key element in the successful deployment of Red Hat OpenShift virtualization is having a robust, flexible, and reliable storage system like Hitachi Virtual Storage System (VSP) that stores different types of workloads, virtual machines, and meets a wide variety of requirements in a highly dynamic environment. VSP One storage with Red Hat OpenShift Virtualization provides a highly available and high-performance environment for virtual machines and container applications.

Using well-known and proven CSI (Container Storage Interface) storage integrations, you can provide persistent storage for virtual machines and stateful container applications.

The integration of Hitachi Storage Plug-in for Containers (HSPC) with OpenShift brings other benefits such as snapshot and cloning and restore operations for persistent volumes, enabling rapid copy creation for immediate use in decision support, software development, and data protection operations.

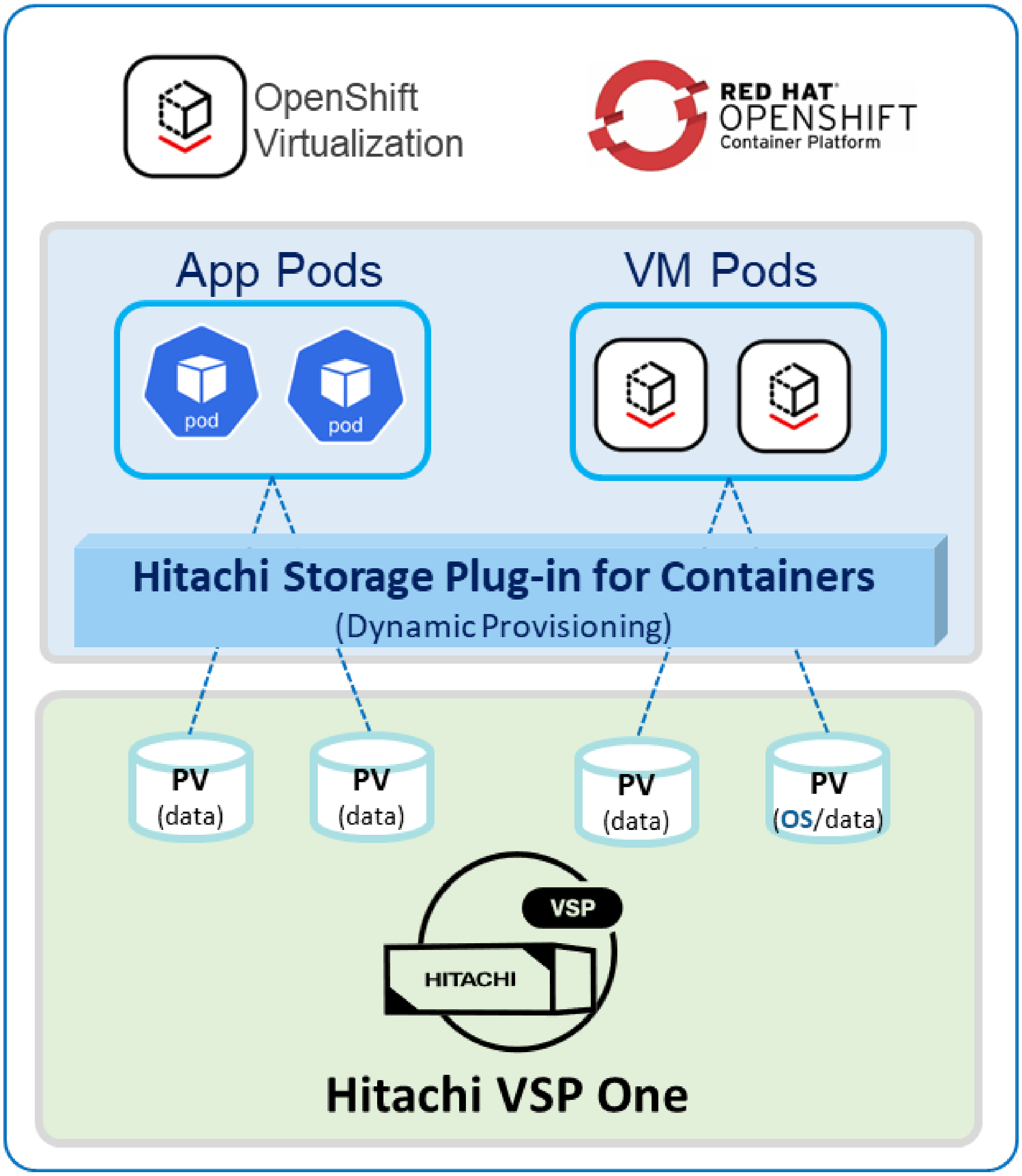
Red Hat OpenShift backed by VSP One storage gives you the peace of mind on the integrations that are needed for your organization to successfully provide workloads, including virtual machines and container services, to your application teams.

This reference architecture also provides the reference design for a build-your-own Red Hat OpenShift Container Platform environment using Hitachi Virtual Storage Platform One Block. Although a specific converged system is used as an example, this reference design still applies to building your own container platform.

The intended audience of this document is IT administrators, system architects, consultants, and sales engineers to assist in planning, designing, and implementing VSP storage with OpenShift Container Platform solutions.

Overview

Red Hat OpenShift is a successful container orchestration platform and is one of the container orchestration solutions supported by VSP storage. The following figure shows a high-level diagram of Red Hat OpenShift Container Platform with OpenShift Virtualization managing virtual machines alongside other containerized application, using Hitachi Virtual Storage Platform One for persistent storage, and using Hitachi Storage Plug-in for Containers (HSPC).



OpenShift Virtualization leverages the RHEL KVM hypervisor and supports running virtual machines in container and managed as Pods. It allows the VM to be managed by Kubernetes and KubeVirt.

Now organizations can have a single platform to run and manage not only containerized applications, but also virtual machines. In addition, Red Hat also supports migration of VMs from other source providers like VMware vSphere, Red Hat Virtualization, OpenStack, and other remote OpenShift Virtualization clusters.

Hitachi Virtual Storage Platform One provides a REST API for Hitachi Storage Plug-in for Containers to dynamically provision persistent volumes. The persistent volumes are provided by Virtual Storage Platform-hosted LUNs through a block protocol to the worker nodes.

- Hitachi Storage Plug-in for Containers (HSPC) dynamically provisions persistent volumes for stateful containers from Hitachi storage.
- This Hitachi CSI driver includes support for `ReadWriteMany (RWX)` access mode which is required to support live migration of virtual machines across cluster nodes.

Follow the steps in Solution design and Solution Implementation and Validation to learn about these new capabilities when using VSP storage with Red Hat OpenShift Container Platform and OpenShift Virtualization.

Accelerate virtual machine migration to OpenShift with storage offload

Hitachi has partnered with Red Hat to introduce a powerful storage offload feature in the Migration Toolkit for Virtualization (MTV) Operator, available starting with MTV version 2.9 as Tech Preview. If you are planning to migrate VM workloads from a VMware vSphere cluster to OpenShift Virtualization, and both environments are backed by the same VSP One storage system, you can take advantage of this feature to dramatically streamline the migration process.

Learn more about this collaboration and feature from the following article:

https://www.hitachivantara.com/en-us/blog/replatform-faster-openshift-vsp-one-storage-offload

Refer to the latest MTV documentation for specifications and details:

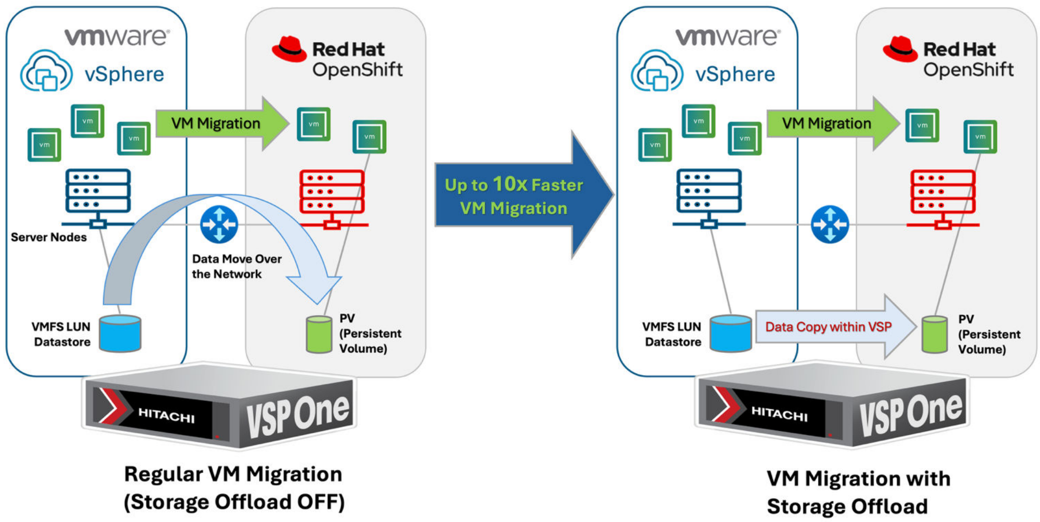
https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/

The following lists some of the main benefits of this storage offload migration:

- Up to 10 × faster migration: Internal testing shows up to 90% reduction in migration time – reducing a 10-hour process into a 1-hour operation.
- No IP network dependency: VM volume data never leaves the VSP storage, freeing up network bandwidth and reducing latency.
- Minimizes compute host resource usage: Less time spent on migration tasks, preserving CPU and memory for critical workloads.

Note: As of MTV version 2.9, the storage offload feature is available for cold migrations only.

The following diagram illustrates VM migration with the storage offload feature.



The following details the difference between having the storage offload feature OFF and ON during VM migration:

- For regular VM migration with storage offload OFF:
  - A VM migration plan is created and completed using Migration Toolkit for Virtualization (MTV) Operator on OpenShift.
  - An empty persistent volume (PV) is provisioned from the VSP storage for the destination VM in OpenShift cluster.
  - From the ESXi host, the target VM data (VMDK) is read from VMFS datastore (backed by the VSP storage).
  - The data is transferred across the network to the OpenShift cluster.
  - The data is written into the persistent volume attached to the new VM.
- For VM migration with storage offload ON:
  - A VM migration plan is created and completed using Migration Toolkit for Virtualization (MTV) Operator on OpenShift.
  - An empty persistent volume (PV) is provisioned from the VSP storage for the destination VM in the OpenShift cluster.
  - The VM data is copied directly within the VSP storage using XCOPY command (bypassing the network):
  - The same PV is temporarily attached to the source ESXi host.
  - The storage offload XCOPY command is issued from the ESXi host to copy the VM data from the VMFS datastore to the newly attached PV.
  - After the data copy is complete, the PV is detached from the ESXi host.

For detailed implement procedures, see Migrate virtual machines with storage offload (on page 48).

Solution components

This section outlines the components used in this reference architecture.

Hitachi hardware components

The tested solution used specific features based on the following hardware. You can use any qualified server platform such as Hitachi Advanced Server.

Hardware	Description	Version	Quantity
Hitachi Advanced Server HA800 series (for VMware compute cluster)	▪ 2 × Intel(R) Xeon(R) Gold 6454S processors ▪ 32 × 32 GB DIMM, 1 TB memory ▪ NS204-u RAID1 for boot ▪ SN1610E 32 Gb 2p FC HBA ▪ 2 × Intel(R) Eth E810-XXVDA2 NICs ▪ 2 × SR932i-p controllers ▪ vSAN Cache Tier: 1 × 800 GB SAS 24G MU SFF ▪ vSAN Capacity Tier: 4 × 1.92 TB SAS 24G RI SFF	iLO 6 BIOS: BIOS: U54 v1.46	3
Hitachi Advanced Server HA800 series (for OCP bare metal worker noes)	▪ 2 × Intel(R) Xeon(R) Gold 6454S processors ▪ 16 × 32 GB DIMM, 512 GB memory ▪ NS204-u RAID1 for boot ▪ SN1610E 32 Gb 2p FC HBA ▪ 2 × Intel(R) Eth E810-XXVDA2 NICs	iLO 6 BIOS: BIOS: U54 v1.46	2
Hitachi Virtual Storage Platform E1090 (for both VMware and OCP clusters)	▪ 2 TB cache ▪ 16 ×3.8 TB NVMe drives ▪ 4 × 32 Gbps Fibre Channel ports	93-07-21/00	1
Hitachi Virtual Storage Platform One Block 28 (for OCP cluster)	▪ 1 TB cache ▪ 24 ×3.8 TB NVMe drives ▪ 4 × 32 Gbps Fibre Channel ports	A3-02-01-40/00	1

Hardware	Description	Version	Quantity
VSP One SDS Block (for OCP cluster)	▪ 26 TB total capacity ▪ 36 × 800 GB SAS SSD drives ▪ iSCSI	1.13	1
Cisco Nexus 93180YC-FX3 switch (leaf)	▪ 48-port10/25 GbE ▪ 6-port40/100 GbE	NXOS 10.3(4a)	2
Cisco Nexus 92348	▪ 48-port 1 GbE ▪ 4-port1/10/25 GbE ▪ 2-port40/100 GbE	NXOS 9.3.(8)	1
Brocade G720	▪ 48-port16/32 Gbps Fibre Channel switch	9.1.1c	2

Software components

The following table lists the key software components.

Software	Version
Hitachi Storage Virtualization Operating System RF	93-07-21/00
VSP One SDS Block	1.13
Hitachi Storage Plug-in for Containers (HSPC)	3.16.0
Red Hat OpenShift Container Platform (OCP)	4.18
OpenShift Virtualization Operator (OCP-V/ Kubevirt)	4.18.13
Migration Toolkit for Virtualization Operator (MTV)	2.9.2
VMware vSphere	8.0 U2 or newer
VMware Virtual Disk Development Kit (VDDK)	8.0 U2
Windows Server VMs	—
Linux VMs	RHEL 9

Red Hat OpenShift Container Platform

Red Hat OpenShift Container Platform (OCP) provides an integrated system to build, deploy, and manage applications consistently across on-premises and hybrid cloud deployments. OCP provides the control plane and data plane within the same interface. OCP provides administrator views to deploy operators, monitor container resources, manage container health, manage users, work with operators, manage pods and deployment configurations, as well as define storage resources.

OCP also provides a developer view to allow users to deploy application resources from various pre-defined resources such as YAML files, Docker files, Catalogs, or GIT within user-defined namespaces. With OCP `kubectl`, a native binary of Kubernetes is complemented by the `oc` command, which provides further support for OCP resources, such as deployment and build configurations, routes, image streams, and tags. OCP provides a GUI and a CLI interface.

Red Hat OpenShift Virtualization

OpenShift Virtualization is a feature of Red Hat OpenShift Container Platform (OCP) that allows you to run virtual machines running in containers and can be managed as native Kubernetes objects. OpenShift Virtualization uses KVM, the Linux kernel hypervisor.

OpenShift Virtualization enables the following virtualization tasks:

- Creating and managing Linux and Windows virtual machines (VMs)
- Running VM workloads alongside pods in the same cluster
- Importing virtual machines from VMware vSphere, KVM, OpenStack, and other environments
- Cloning virtual machines
- Live migrating of virtual machines between the nodes

Migration Toolkit for Virtualization

Migration Toolkit for Virtualization (MTV) enables you to migrate virtual machines from different sources providers to an OpenShift Virtualization destination provider. The following source providers are supported:

- VMware vSphere and Open Virtual Appliances (OVAs) created by VMware vSphere
- Red Hat Virtualization (RHV)
- OpenStack
- Remote OpenShift Virtualization clusters

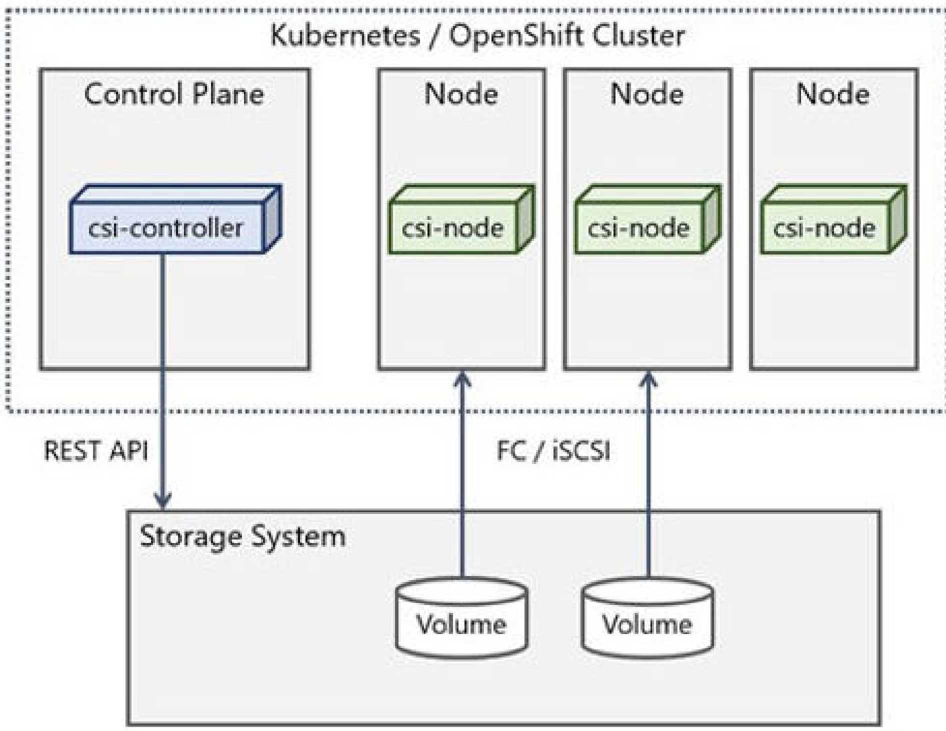
This paper covers migration of virtual machines from VMware vSphere to OpenShift Virtualization using VSP One Block for persistent storage.

Hitachi Storage Plug-in for Containers

Hitachi Storage Plug-in for Containers (HSPC) is a software component that contains libraries, settings, and commands that you can use to create a container to run stateful applications. It enables stateful applications to persist and maintain data after the lifecycle of the container has ended. Storage Plug-in for Containers provides persistent volumes from Hitachi Dynamic Provisioning (HDP) or Hitachi Thin Image (TI) pools to bare metal or hybrid deployments using Fibre Channel, NVMe over Fibre Channel, or iSCSI protocols. iSCSI protocol is supported for both bare metal and virtual environments.

Storage Plug-in for Containers integrates Kubernetes or OpenShift with Hitachi storage systems using Container Storage Interface (CSI).

The following diagram illustrates a container environment where Storage Plug-in for Containers is deployed.



Volume and access modes for virtual machines

When running virtual machines in a containerized environment such as Red Hat OpenShift, it is critical to use the right storage provider. The Hitachi HSPC CSI driver supports `ReadWriteMany` (RWX) access mode and block volume. This is required to support live migration of virtual machines across cluster nodes.

When you deploy Virtual Machines or migrate VMs from another source provider such as VMware, the Virtual Machines will automatically be created with Persistent Volume Claims (PVCs) with a shared `ReadWriteMany` (RWX) access mode. No additional setting is required at the Storage Profile or Storage Classes.

Volume snapshots

In OpenShift or Kubernetes, creating a PersistentVolumeClaim (PVC) initiates the creation of a PersistentVolume (PV), which contains the data. A PVC also specifies a StorageClass, which provides additional attributes for backend storage.

Because this guide also covers snapshots of PVCs, it is important to clarify some additional concepts related to snapshots. A VolumeSnapshot represents a snapshot of a volume on the storage system. In the same way API resources PersistentVolume and PersistentVolumeClaim are used to provision volumes for users and administrators, VolumeSnapshot and VolumeSnapshotContent API resources are provided to create volume snapshots. VolumeSnapshot support is only available for CSI drivers.

- `VolumeSnapshotContent` — Represents a snapshot taken of a volume in the cluster. Similar to the PersistentVolume object, the VolumeSnapshotContent is a cluster resource that points to a real snapshot in the backend storage. VolumeSnapshotContent is not namespaced.
- `VolumeSnapshot` — Is a request for a snapshot of a volume. It is similar to a PersistentVolumeClaim. Creating a VolumeSnaphot triggers a snapshot (VolumeSnapshotContent), and the objects are bound together. There is a one-to-one binding between VolumeSnaphot and VolumeSnapshotContent. VolumeSnapshot is namespaced.
- `VolumeSnapshotClass` — Allows you to define different attributes belonging to a VolumeSnapshot. This is similar to how a StorageClass is used for PVs.

This is covered in *Cloning and snapshots of Virtual Machines* as a requirement for CSI snapshots.

> **Note:** In addition to snapshots, HSPC supports volume cloning and volume expansion features. For details see the HSPC Reference Guide at https://docs.hitachivantara.com/search/all?query=Hitachi+Storage+Plug-in+for+Containers&content-lang=en-US.

> **Note:** From a migration and replication services perspective, Hitachi Replication Plug-in for Containers (HRPC) provides replication data services for the persistent volumes on Hitachi Virtual Storage Platform (VSP). With HRPC, persistent volumes can be snapshotted and cloned locally or to remote Kubernetes clusters with their own remote VSP storage platform. In addition, HSPC has a Technology Preview for the Stretched PersistentVolumeClaim (PVC) feature that automates the provisioning of synchronous replication between the storage systems at each site in a single Kubernetes or OpenShift cluster that spans two sites. For more details, see https://docs.hitachivantara.com/search/all?query=Hitachi+Replication+Plug-in+for+Containers&content-lang=en-US and https://docs.hitachivantara.com/search/all?query=Hitachi+Storage+Plug-in+for+Containers&content-lang=en-US.

HSPC and VSP Resource Groups

You can partition storage system resources by limiting the LDEV ID range added to the resource group for a specific Kubernetes cluster. You can also isolate impacts between Kubernetes clusters. The following requirements should be met:

- Multiple Kubernetes clusters can share one resource group.
- Storage system users must have access only to the resource group that they created. The storage system user must not have access to other resource groups..
- Create a pool from pool volumes with the resource group that you have created.
- Allocate the necessary number of undefined LDEV IDs to the resource group.
- Allocate the necessary number of undefined host group IDs to the resource group for each storage system port defined in StorageClass. The number of host group IDs must be equal to the number of hosts for all ports.

For details, see the Hitachi Storage Plug-in for Containers documentation at https://docs.hitachivantara.com/search/all?query=Hitachi+Storage+Plug-in+for+Containers&content-lang=en-US.

Solution design

This section outlines the detailed solution example for Red Hat OpenShift and Red Hat OpenShift Virtualization powered by VSP One storage.

Solution considerations

- Size an OpenShift cluster based on the number of virtual machines, the size of the VMs, specifically CPU and memory, the amount of overhead for the VMs, and other hosted applications. For additional details see the *OpenShift Virtualization - Reference Implementation Guide* at https://access.redhat.com/articles/7067871 or contact Hitachi Vantara professional services for guidance.
- Account for enough resources for failover/HA and resource balancing and consider resources to accommodate that capacity in the event of a failure scenario — or even when taking nodes offline, for example, to perform cluster updates and upgrades.
- On the network side, consider network throughput for cluster functions, SDN, live migration, storage traffic, and hosted applications.
- You can choose to have one or more clusters for special workloads or for VMs that require a considerable amount of resources or use features such as taints and node selectors.
- On the storage side, if you are using multiple clusters, consider using the resource partitioning function provided by Hitachi Storage Plug-in for Containers together with Hitachi Virtual Storage Platform One Block storage. With HSPC you can partition storage system resources and allocate resources to specific Kubernetes clusters, and this way you can isolate the impact between Kubernetes clusters.

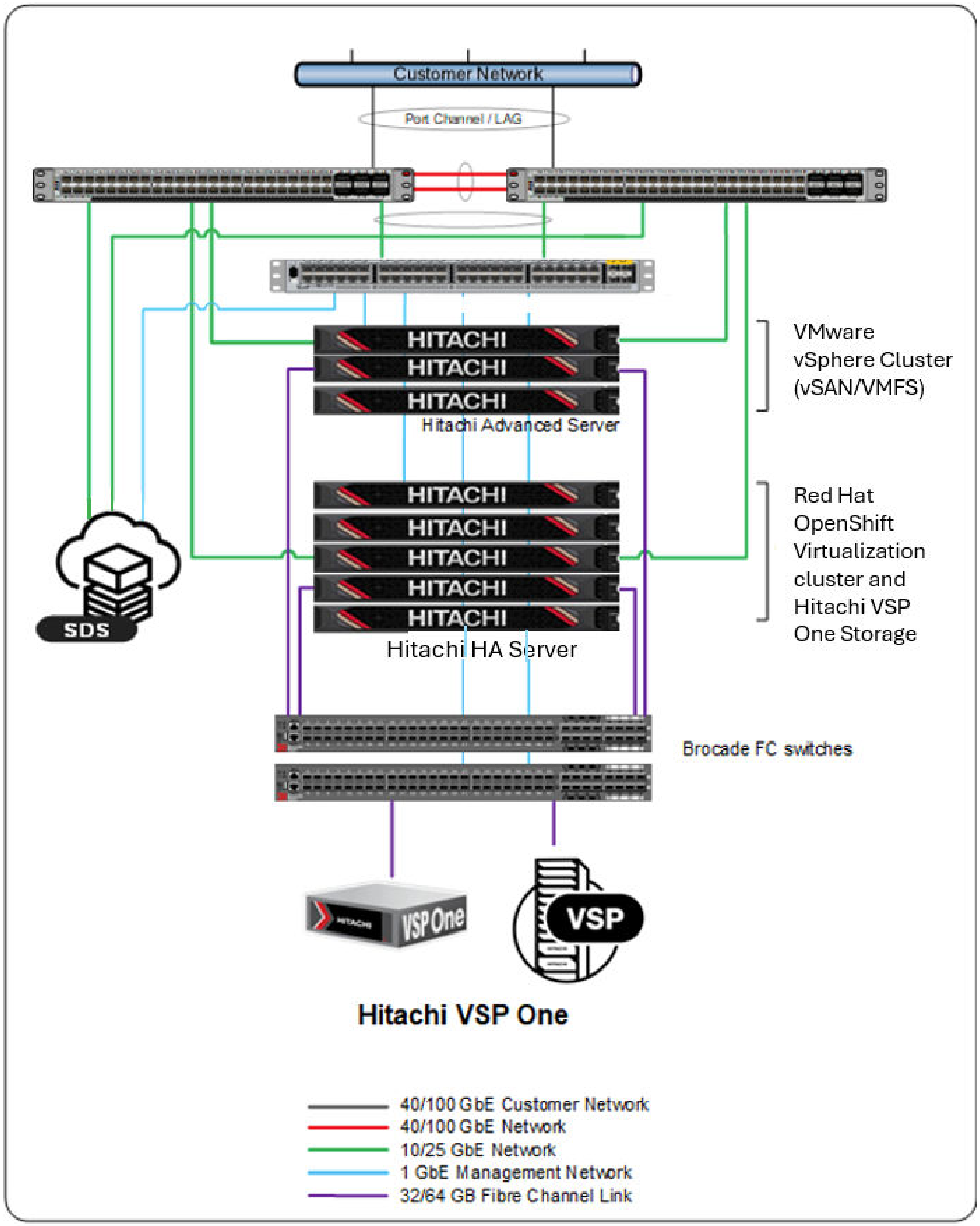
For details, see Resource Partitioning in the *HSPC Reference Guide* at https://docs.hitachivantara.com/search/all?query=Hitachi+Storage+Plug-in+for+Containers&content-lang=en-US.

Infrastructure components

The following figure shows a high-level architecture of the infrastructure and clusters used to validate the Red Hat OpenShift Virtualization solution with VSP storage. It includes the following components:

- Hitachi HA servers for Red Hat OpenShift Container Platform and OpenShift Virtualization:
  - The OpenShift clusters have 3 × virtual control-planes that are hosted by VMware vSphere and 2 × Hitachi HA800 series servers that serve as bare metal worker nodes configured to run workloads.
  - Three Hitachi Virtual Storage Platform storage systems for persistent storage for standard Pods and VM Pods (current generation VSP 5000 series or VSP E series storage systems can be used as well):
    - Two Hitachi VSP One Block storage systems
    - One Hitachi VSP One Block SDS
- VMware vSphere cluster:
  - For VMFS, leverage the HBA PCIe card, which is optionally configured together with Hitachi vSAN Ready Nodes that are formed as a VMware Cloud Foundation environment for access to VSP storage.
  - For vSAN compute nodes, leverage supported internal drives. These Hitachi compute nodes are certified as vSAN clusters.
  - One Hitachi VSP One Block storage system.
- The following network switches are used:
  - Two Cisco 9332C or Arista 7050CX3 spine Ethernet switches
  - Two Cisco 93180YC-FX3 or Arista 7050SX3 leaf Ethernet switches
  - One Cisco 92348 or Arista 7010T management switch

The following diagram represents the high-level architecture that was used for this reference architecture.

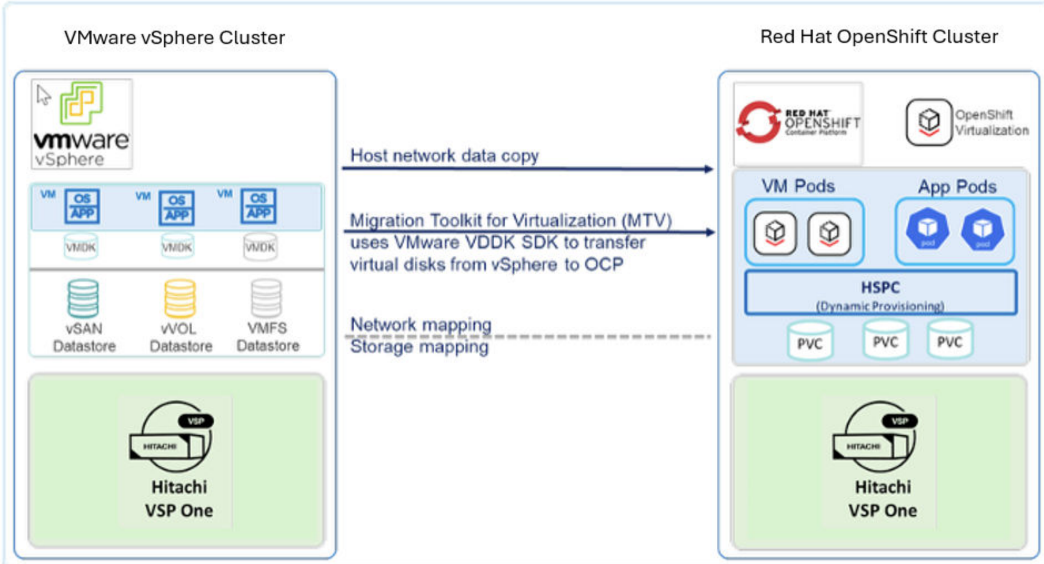


Deploy OpenShift Container Platform

This guide does not cover the step-by-step details of how to implement OCP. Follow Red Hat OCP documentation for the setup of the cluster.

One OCP cluster has been configured to support the different use cases related to virtualization as described in this guide. In addition, a VMware cluster has been configured to validate migration of virtual machines from the VMware to OpenShift cluster.

The following figure illustrates a hybrid OpenShift Container Platform architecture using VSP one storage systems for both clusters.

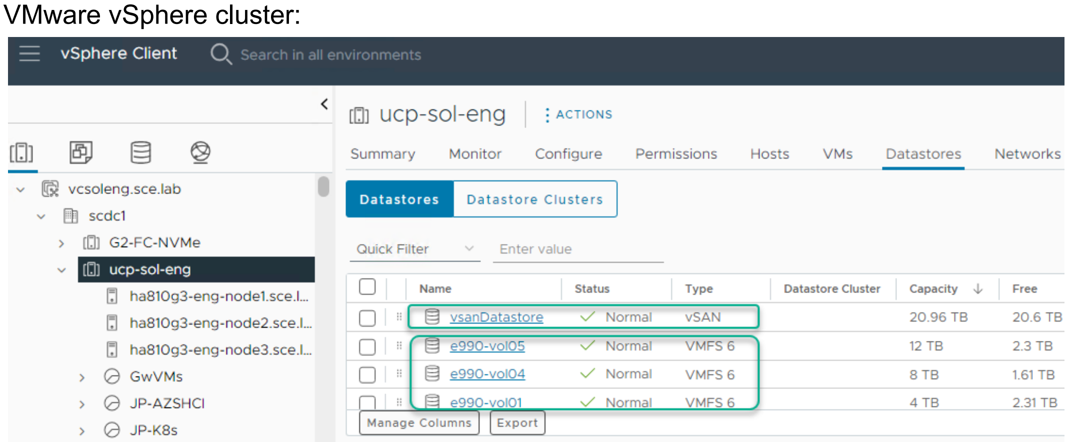


The following are additional details about the two clusters.

- OpenShift Container Platform hosts OpenShift Virtualization:
  - Red Hat OpenShift v.4.17
  - Kubernetes v1.30.11
  - 3 × virtual control-planes
  - 2 × physical Hitachi HA810 worker nodes
  - Hitachi storage systems:
    - VSP E1090
    - VSP One Block 28
    - VSP One SDS Block

  For detailed OpenShift Container Platform installation procedures, see the Red Hat documentation https://docs.redhat.com/en.

- VMware vSphere cluster:



Deploy Hitachi Storage Plug-in for Containers

Hitachi Storage Plug-in for Containers (HSPC) is easily deployed to OpenShift using the Operator, which can be installed from OperatorHub.

This guide does not cover the step-by-step how to install HSPC, follow the Hitachi Storage Plug-in for Containers Quick Reference Guide or the steps described in the Hitachi Storage Integrations with Red Hat OpenShift Reference Architecture.

In summary, these steps include:

- Install Hitachi Storage Plug-in for Containers.
- Configure Secret settings to access VSP One storage systems.
- Configure StorageClass settings.
- Configure Multipathing (Fibre Channel or iSCSI).

Specific steps for how to configure Storage Classes will be covered as part of each of the use cases in the Solution implementation and validation section.

> **Note:** If there is a previous version of Storage Plug-in for Containers, remove it before performing the installation procedure.

HSPC and VSP Host Groups

Host groups required for Storage Plug-in for Containers (HSPC) are automatically created by HSPC. It automatically searches host groups and iSCSI targets based on the name.

To use existing host groups, rename them according to the naming rule. For details, see *Host group and iSCSI target naming rules* in the *Hitachi Storage Plug-in for Containers Quick Reference Guide* at https://docs.hitachivantara.com/v/u/en-us/adapters-and-drivers/3.15.x/mk-92adptr142.

> **Note:** Storage Plug-in for Containers (HSPC) will overwrite host mode options even if existing host groups have other host mode options.

Red Hat OpenShift Virtualization with Hitachi VSP One storage

Prerequisites

- Red Hat OpenShift cluster includes bare metal worker nodes:
  - You must have multiple worker nodes at the time of installation if you want to use live migration features.
  - Requirements for live migration:
    - Make sure to have shared storage.
    - The CSI provider must support `ReadWriteMany (RWX)` access mode. HSPC comes with `ReadWriteMany` support.
- You can use any of the four installation methods (user-provisioned, installer-provisioned, assisted, agent-based installer) of an OCP cluster on bare metal.
- Use Hitachi Virtual Storage Platform One Block as the backend storage to the OCP cluster and OpenShift Virtualization.
- Hitachi Storage Plug-in for Containers (HSPC).
- A `storageClass` defined on the OCP cluster with HSPC as the provisioner.
- When planning for cluster resources, make sure to account for enough additional CPU, memory, and storage to support the additional overhead imposed by the OpenShift Virtualization feature. Follow Red Hat documentation to calculate the overhead of these resources.
- Always plan your environment according to the tested object maximums.
  - OpenShift Container Platform object maximums
  - OpenShift Virtualization object maximums

Supported guest operating systems

To view the supported guest operating systems for OpenShift Virtualization, see Certified Guest Operating Systems in Red Hat OpenStack Platform, Red Hat Virtualization, OpenShift Virtualization and Red Hat Enterprise Linux with KVM.

Install and configure OpenShift Virtualization

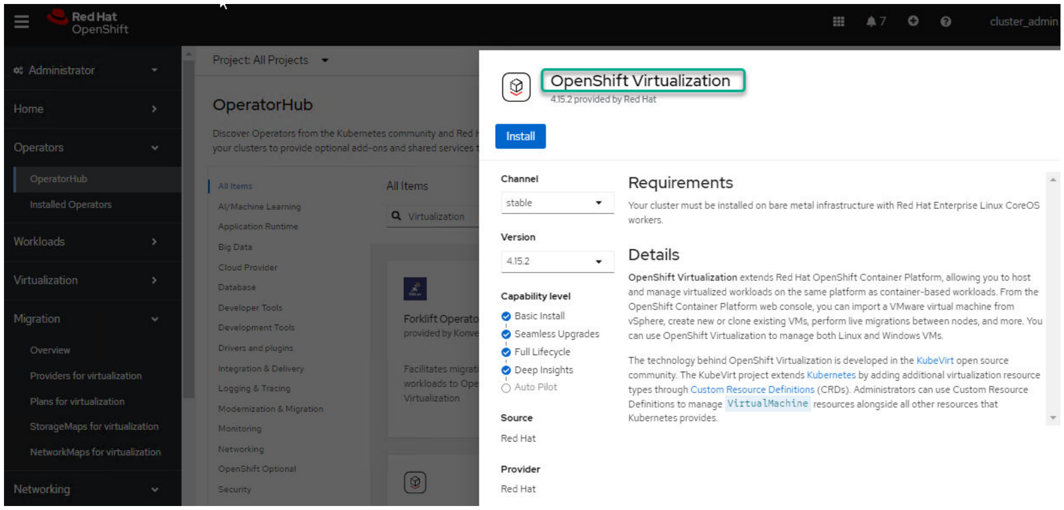
OpenShift Virtualization Operator can be deployed from the web console or the CLI. This operator includes the Virtualization plugin for the OCP web console. This reference architecture describes the process from the web console using the OperatorHub.

To deploy the OpenShift Virtualization Operator from the web console, follow these steps.

Procedure

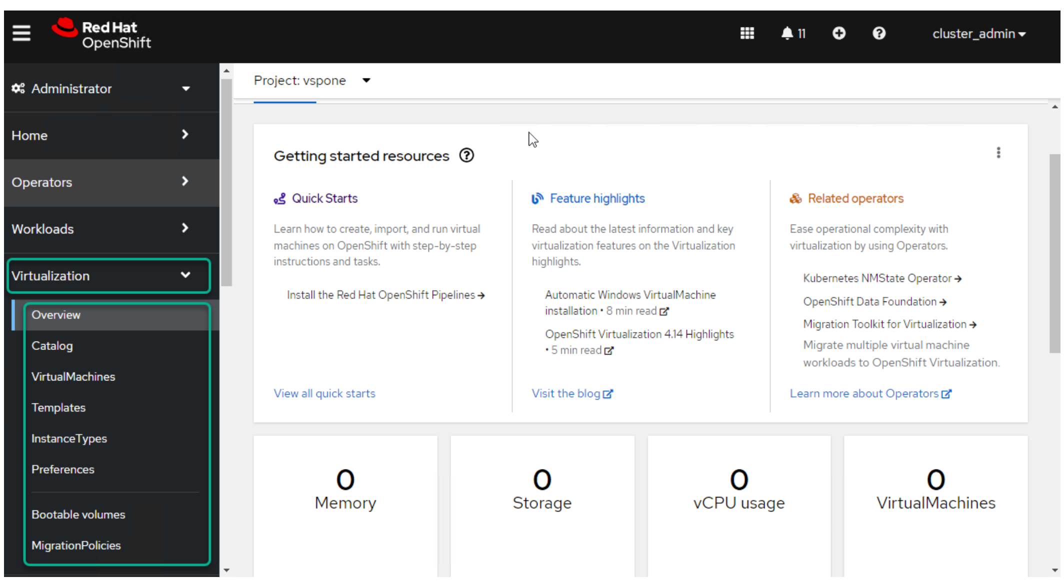
1. Log in to the Red Hat OCP web console as a user with cluster-admin permissions.
2. Navigate to **`Operators`** > **`OperatorHub`** and filter by the keyword **`Virtualization`**

3. Select the OpenShift Virtualization Operator tile with the Red Hat source label and click **`Install`**.



4. On the **`Install Operator`** page, leave the default parameters and click **`Install`** to make the Operator available on the `openshift-cnv` namespace which also will be created during the installation process.
5. Wait for the operator installation to complete, then click **`Create HyperConverged`**.
6. (Optional): Configure Infra and Workloads node placement options.
7. Click **`Create`** to launch OpenShift Virtualization. You might need to refresh the web console to see a new option called **`Virtualization`** on the OCP web console.

   The following figure shows the OpenShift Virtualization user interface integrated into the OCP web console.



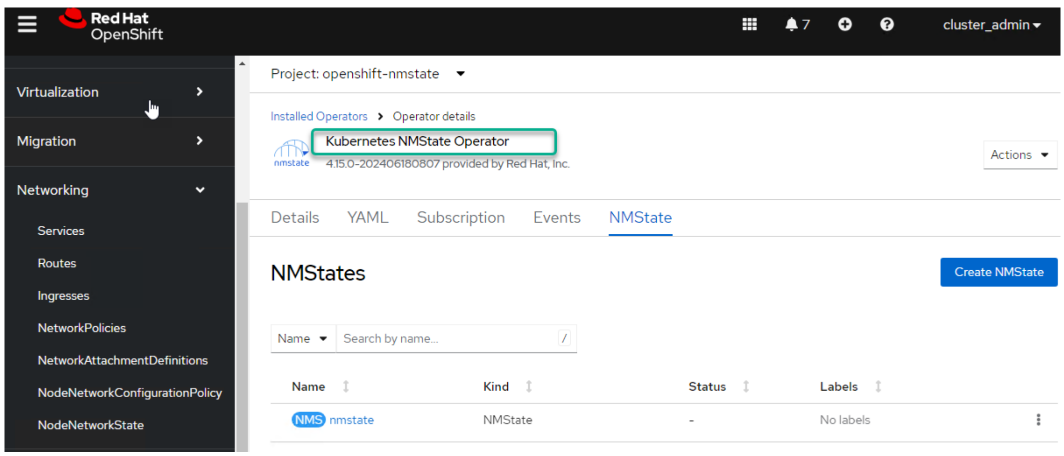
Post installation configurations

After the installation of OpenShift Virtualization, you can configure the following components depending on your environment.

- Configure node placement rules. See Specifying nodes for OpenShift Virtualization components for additional details.
- Network configurations:
  - Install the Kubernetes `NMState` Operators

    NMState allows you to configure a Linux Bridge for live migration or external access to virtual machines.
  - Install the `SR-IOV` Operators

    The Single Root I/O Virtualization (SR-IOV) specification is a standard for a type of PCI device assignment that can share a single device with multiple pods. This operator allows you to manage SR-IOV network devices and network attachments.
  - See Post installation network configuration for additional details.



  - (Optional) Enable the creation of load balancers services using the OCP web console.

- Storage configurations:
  - Configure a default `storageClass` for your cluster leveraging the Hitachi Storage Plug-in for Containers (HSPC) CSI driver. An example is provided in the HSPC section.

```
[root@jputilitysrv1 ocpjpcl1]# oc get sc
NAME                          PROVISIONER            RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION
vspone-block28-205-sc         hspc.csi.hitachi.com   Delete          Immediate           true
vspone-e1090-117-sc (default) hspc.csi.hitachi.com   Delete          Immediate           true
vspone-sdsb-55-sc             hspc.csi.hitachi.com   Delete          Immediate           true
```

    To make a storageClass as default run the following command:

```
oc patch storageclass sc-vsp1b28 -p '{"metadata":
{"annotations": {"storageclass.kubernetes.io/is-default-class":
"true"}}}'
```

    The following is an example of one of the storage classes `vspone-e1090-117-sc` defined for VSP One Block storage.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  annotations:
    kubernetes.io/description: Hitachi Storage Plug-in for Containers
  name: vspone-e1090-117-sc
parameters:
  serialNumber: "715021"
  poolID: "2"
  portID: CL5-A,CL6-A
  connectionType: fc
  csi.storage.k8s.io/fstype: ext4
  csi.storage.k8s.io/controller-expand-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/controller-expand-secret-namespace: vspone
  csi.storage.k8s.io/controller-publish-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/controller-publish-secret-namespace: vspone
  csi.storage.k8s.io/node-publish-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/node-publish-secret-namespace: vspone
  csi.storage.k8s.io/node-stage-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/node-stage-secret-namespace: vspone
  csi.storage.k8s.io/provisioner-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/provisioner-secret-namespace: vspone
provisioner: hspc.csi.hitachi.com
reclaimPolicy: Delete
volumeBindingMode: Immediate
allowVolumeExpansion: true
```

  - It is also mandatory to configure storage profiles. However, with the latest HSPC there is no need for additional configuration because the Storage Profile will be automatically configured with the recommended storage settings based on the associated storage class.

```
[root@jputilitysrv1 ocpjpcl1]# oc get storageprofiles
NAME                       AGE
vspone-block28-205-sc      4d1h
vspone-e1090-117-sc        13d
vspone-sdsb-55-sc          13d
[root@jputilitysrv1 ocpjpcl1]#
```

Here we can see that one storage profile has been created automatically for every storage class, and the values on the storage profile are pre-configured by the Hitachi HSPC provider.

> **Note:** From OpenShift Virtualization 4.15, it is possible to define the snapshotClass in the storage profile. This allows association of a particular volumesnapshotclass to storageClass. You can edit the storage profile and provide the required *volumesnapshotclass* name. This is important because designs for Hitachi VSP typically involve multiple storage pools. For additional details see Red Hat documents https://access.redhat.com/solutions/7036331 or https://docs.openshift.com/container-platform/4.15/rest_api/storage_apis/volumesnapshot-snapshot-storage-k8s-io-v1.html.

- The command oc get storageprofile <storage profile name> -oyaml can be used to verify the config and key values such as ReadWriteMany (access mode required for live migration) for one of the storage profiles vspone-block28-205-sc which is associated with storage class vspone-block28-205-sc.

```
[root@jputilitysrv1 ocpjpcl1]# oc get storageprofile vspone-block28-205-sc -oyaml
apiVersion: cdi.kubevirt.io/v1beta1
kind: StorageProfile
metadata:
  labels:
    app.kubernetes.io/managed-by: cdi-controller
    app.kubernetes.io/part-of: hyperconverged-cluster
    ...
  name: vspone-block28-205-sc
  ownerReferences:
  - apiVersion: cdi.kubevirt.io/v1beta1
    blockOwnerDeletion: true
    controller: true
    kind: CDI
    name: cdi-kubevirt-hyperconverged
...
spec: {}
status:
  claimPropertySets:
  - accessModes:
    - ReadWriteMany
    volumeMode: Block
  - accessModes:
    - ReadWriteOnce
    volumeMode: Block
  - accessModes:
    - ReadWriteOnce
    volumeMode: Filesystem
  cloneStrategy: csi-clone
  dataImportCronSourceFormat: pvc
  provisioner: hspc.csi.hitachi.com
  storageClass: vspone-block28-205-sc
```

- For worker nodes connected to VSP One Block storage from Fibre Channel or iSCSI, it is recommended to enable multipathing.

Solution implementation and validation – use cases

This reference architecture was validated by the following:

- Deploying Virtual Machines (VMs) from existing templates using the web console.
- Exploring additional ways to deploy VMs using snapshots, cloning PVCs, or cloning VMs.
- Showing how to expose a VM from a service or connecting a VM to a Linux bridge.
- Performing a live migration of VMs across the worker nodes.
- Migrating VMs from a VMware vSphere environment to OpenShift Virtualization.

Deployment of VMs on OpenShift Virtualization

Red Hat OpenShift Virtualization uses the Kubernetes PersistentVolume (PV) paradigm.

Containerized VMs deployed on OpenShift Virtualization have the same characteristics as non-containerized VMs. They have similar resource limitations (CPU and memory) dictated by the KVM hypervisor.

On the network side, they inherit the pod network by default, and you can use networking operators to configure additional networks.

On the storage side, VMs use the Kubernetes persistent storage paradigm (PVC, PV, and StorageClass) for VM disks (boot and data). These VM disks are backed by persistent storage on Hitachi Virtual Storage Platform One Block and dynamically provisioned by Hitachi Storage Plug-in for Containers (HSPC).

In addition, VMs inherit features and functions from Kubernetes such as scheduling, high availability, and attaching/detaching resources.

Red Hat OpenShift Virtualization offers different ways to deploy VMs. These options include the following:

- Create a VM from a template using the OCP web console.
- Create a VM from an instance type using the OCP web console.
- Create a VM from a `VirtualMachine` manifest using the OCP web console or the command line.
- Create a VM using existing PVCs or snapshots, or using VM clone operations.

In this reference architecture guide we demonstrate some of these use cases.

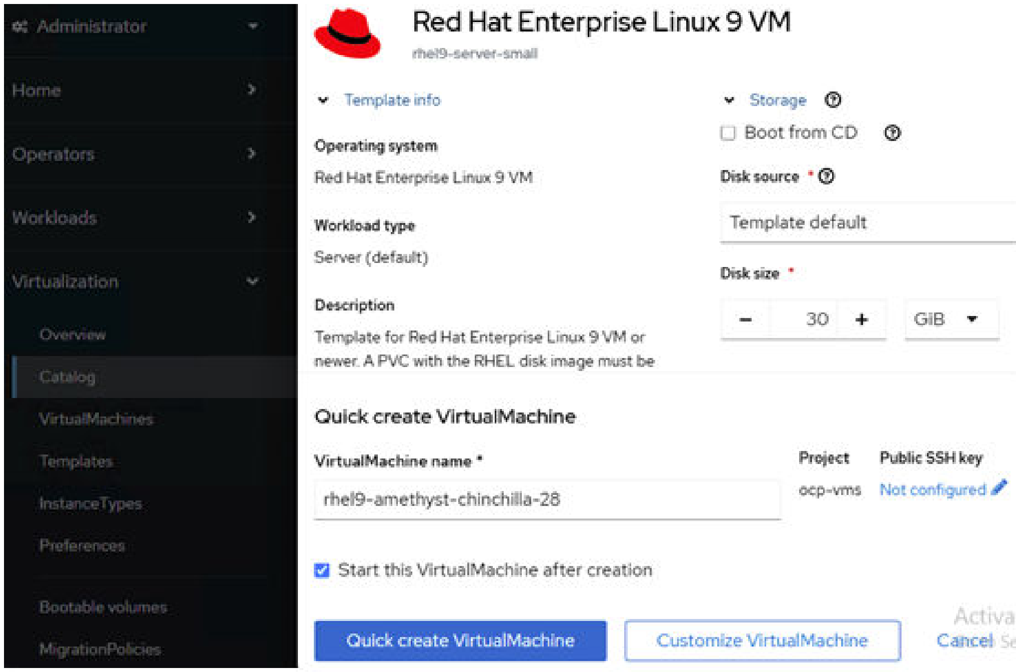
Create a new VM from a template

Use this procedure to create a VM from a template using the web console.

Procedure

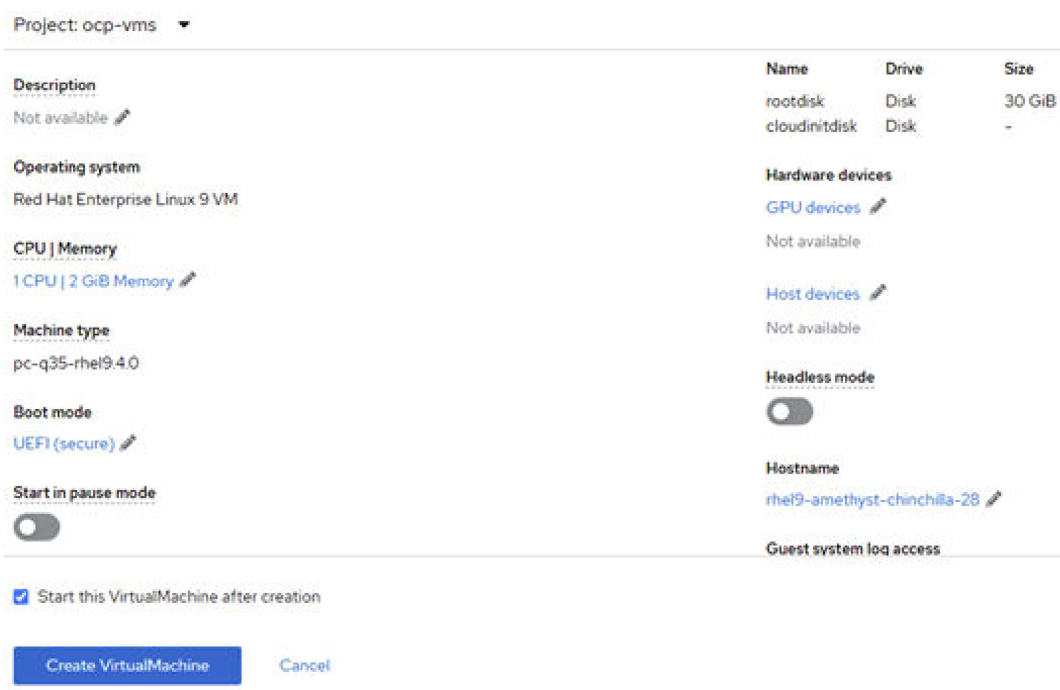
1. Log in to the Red Hat OCP web console.
2. Navigate to **`Virtualization`** > **`Virtual Machines`**.
3. Either create a new project/namespace or select one. Then click **`Create VirtualMachine`**.

   You will be presented with 3 options: From Template, From Volume, or With YAML.
4. Select **`From Template`**, and select one of the templates from the Catalog.
5. Then select either **`Quick create Virtualmachine`** or **`Customize Virtual Machine`**. The following image (RHEL 9) shows the **`Quick create Virtualmachine`** page.



6. Also, you have another option to customize VirtualMachine parameters as the following figure shows.

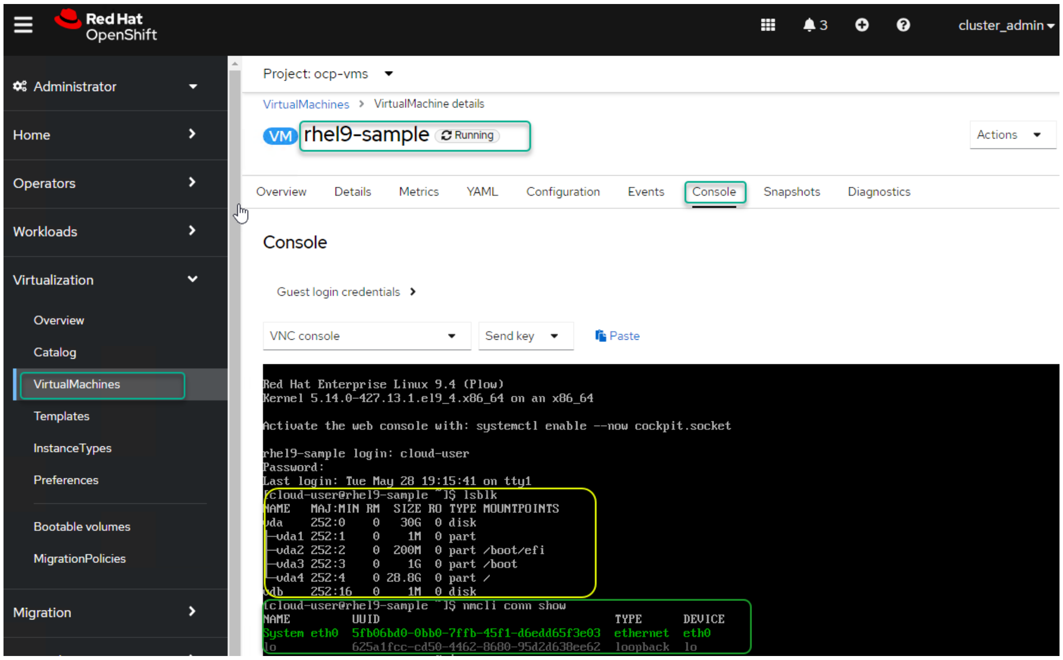
   The **`Customize and create VirtualMachine`** pane displays the Overview, YAML, Scheduling, Environment, Network interfaces, Disks, Scripts, and Metadata tabs.



7. Click **`Create VirtualMachine`**.
8. After the VM is created and in running state, we can log in to the console of the VM and treat it as any non-containerized VM.

Note: The user is **`cloud-user`** and the password is the one you set at the time of VM creation.

Result

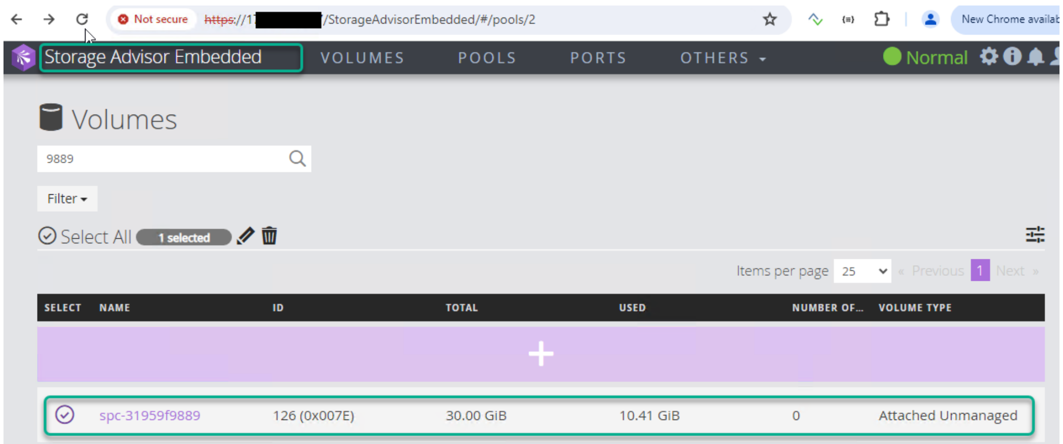


From the CLI, we can use standard `oc` commands to query the VM instances, pods, PV, and PVCs.

```
[root@jputilitysrv1 ocpjpcl1]# oc get vmi
NAME           AGE   PHASE     IP            NODENAME                                  READY
rhel9-sample   26m   Running   10.131.0.101  ocpjpcl1-worker-1.ocpjpcl1.ocp.sce.lab    True
[root@jputilitysrv1 ocpjpcl1]#
[root@jputilitysrv1 ocpjpcl1]# oc get pods
NAME                                READY   STATUS    RESTARTS   AGE
virt-launcher-rhel9-sample-skfcp    1/1     Running   0          26m
[root@jputilitysrv1 ocpjpcl1]#
[root@jputilitysrv1 ocpjpcl1]# oc get pvc
NAME           STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS          AGE
rhel9-sample   Bound    pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917   30Gi       RWX            vspone-e1090-117-sc   26m
[root@jputilitysrv1 ocpjpcl1]#
[root@jputilitysrv1 ocpjpcl1]# oc get pv pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917
NAME                                       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                  STORAGECLASS
pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917   30Gi       RWX            Delete           Bound    ocp-vms/rhel9-sample   vspone-e1090-117-sc
[root@jputilitysrv1 ocpjpcl1]#
```

```
[root@jputilitysrv1 ocpjpcl1]# oc describe pv pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917
Name:            pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917
Labels:          <none>
Annotations:     pv.kubernetes.io/provisioned-by: hspc.csi.hitachi.com
                 volume.kubernetes.io/provisioner-deletion-secret-name: vspone-e1090-secret
                 volume.kubernetes.io/provisioner-deletion-secret-namespace: vspone
Finalizers:      [kubernetes.io/pv-protection external-attacher/hspc-csi-hitachi-com]
StorageClass:    vspone-e1090-117-sc
Status:          Bound
Claim:           ocp-vms/rhel9-sample
Reclaim Policy:  Delete
Access Modes:    RWX
VolumeMode:      Block
Capacity:        30Gi
Node Affinity:   <none>
Message:
Source:
    Type:              CSI (a Container Storage Interface (CSI) volume source)
    Driver:            hspc.csi.hitachi.com
    FSType:
    VolumeHandle:      01--scsi--938000715021--126--spc-31959f9889
    ReadOnly:          false
    VolumeAttributes:      connectionType=fc
                           hostModeOption=
                           ldevIDDec=126
                           ldevIDHex=00:7E
                           nickname=spc-31959f9889
                           ports=CL5-A,CL6-A
                           size=30Gi
                           storage.kubernetes.io/csiProvisionerIdentity=1716937435515-5993-hspc.csi.hitachi.com
Events:                <none>
[root@jputilitysrv1 ocpjpcl1]#
```

And on the Hitachi Virtual Storage Platform One Block storage we can see the corresponding 30Gi volume.



Install the QEMU guest agent and VirtIO drivers

The virtual machines require a guest agent called QEMU that passes information to the host about the VM, users, file systems, and secondary networks. You must install the QEMU guest agent on VMs created from operating system images that are not provided by Red Hat.

Moreover, the QEMU guest agent must be installed if you plan to create snapshots of an online (Running state) VM with the highest integrity. The QEMU guest agent takes a consistent snapshot by attempting to quiesce the VM file system as much as possible, depending on the system workload. This ensures that in-flight I/O is written to the disk before the snapshot is taken. If the guest agent is not present, quiescing is not possible and a best-effort snapshot is taken.

For migrated VMs, MTV automatically embeds the QEMU guest agent into the migrated VM. In Windows you can verify this either on `Programs and Features` or from command line using net start and verify that the output contains QEMU Guest Agent

Follow instructions from https://docs.openshift.com/container-platform/4.15/virt/virtual_machines/creating_vms_custom/virt-installing-qemu-guest-agent.html to properly install or update the QEMU guest agent and VirtIO drivers.

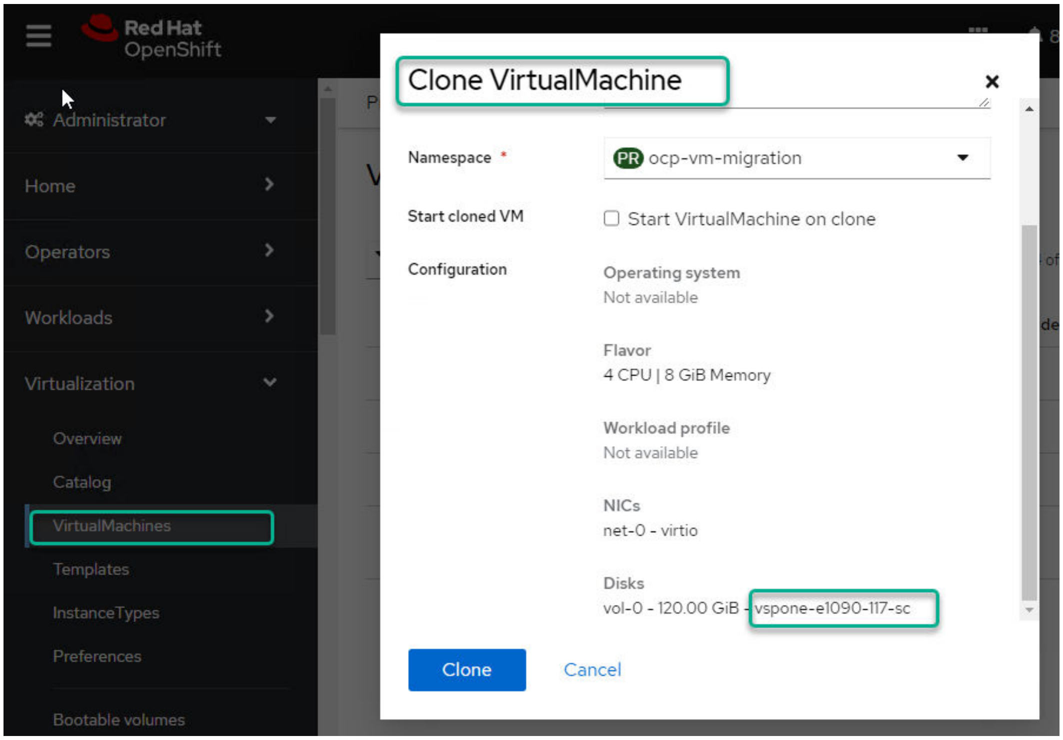
Create new virtual machines using snapshots, cloning PVCs, or cloning VMs

There are 3 new methods to create virtual machines:

- Create a VM using the VM clone operation
- Create a VM from a snapshot
- Create a VM by cloning PVCs

Create a VM using the VM clone operation

This is the easiest and fastest way to create a new VM. Just select the VM you want to clone, click the ellipsis, and select `Clone`. You can change the project, and then click `Clone` as shown.



Create a VM from a snapshot

To deploy a VM from a snapshot, you need to create a snapshot of an existing PVC. One requirement for this is to create a `VolumeSnapshotClass` custom resource (CR) to register the CSI driver. In this example we are using the Hitachi HSPC CSI driver.

The `VolumeSnapshotClass` CR must contain the following parameters:

- The driver must use `hspc.csi.hitachi.com`, which corresponds to Hitachi Storage Plug-in for Containers (HSPC).
- The `poolID` must be the same as the one specified in the `StorageClass`.
- The `secret` name and secret namespace must be the same as the ones specified in the `StorageClass` definition.
- The YAML file below provides an example of a `VolumeSnapshotClass` CR using Hitachi Storage Plug-in for Containers (HSPC).

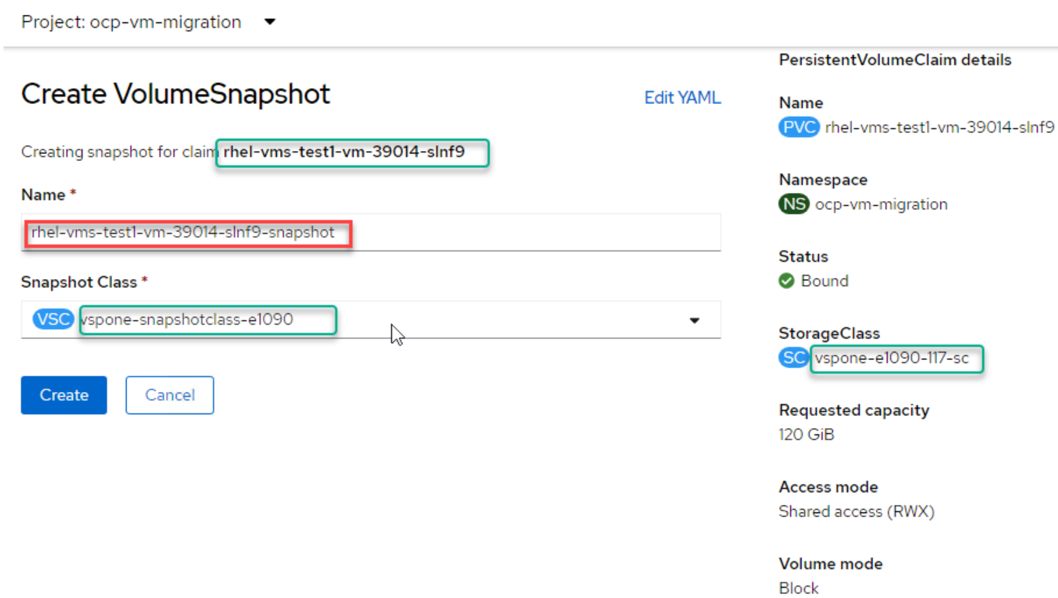
> **Note:** Creating a VolumeSnapshotClass is a one-time operation. All VMs could/would use the same VolumeSnapshotClass or else call out areas where a second snapshotclass would be used.

```
cat vspone-e1090-volumesnapshotclass.yaml
```

```yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: vspone-snapshotclass-e1090
driver: hspc.csi.hitachi.com
deletionPolicy: Delete
parameters:
  poolID: "2"
  csi.storage.k8s.io/snapshotter-secret-name: vspone-e1090-secret
  csi.storage.k8s.io/snapshotter-secret-namespace: vspone
```

To create the VolumeSnapshotClass, run the following command.

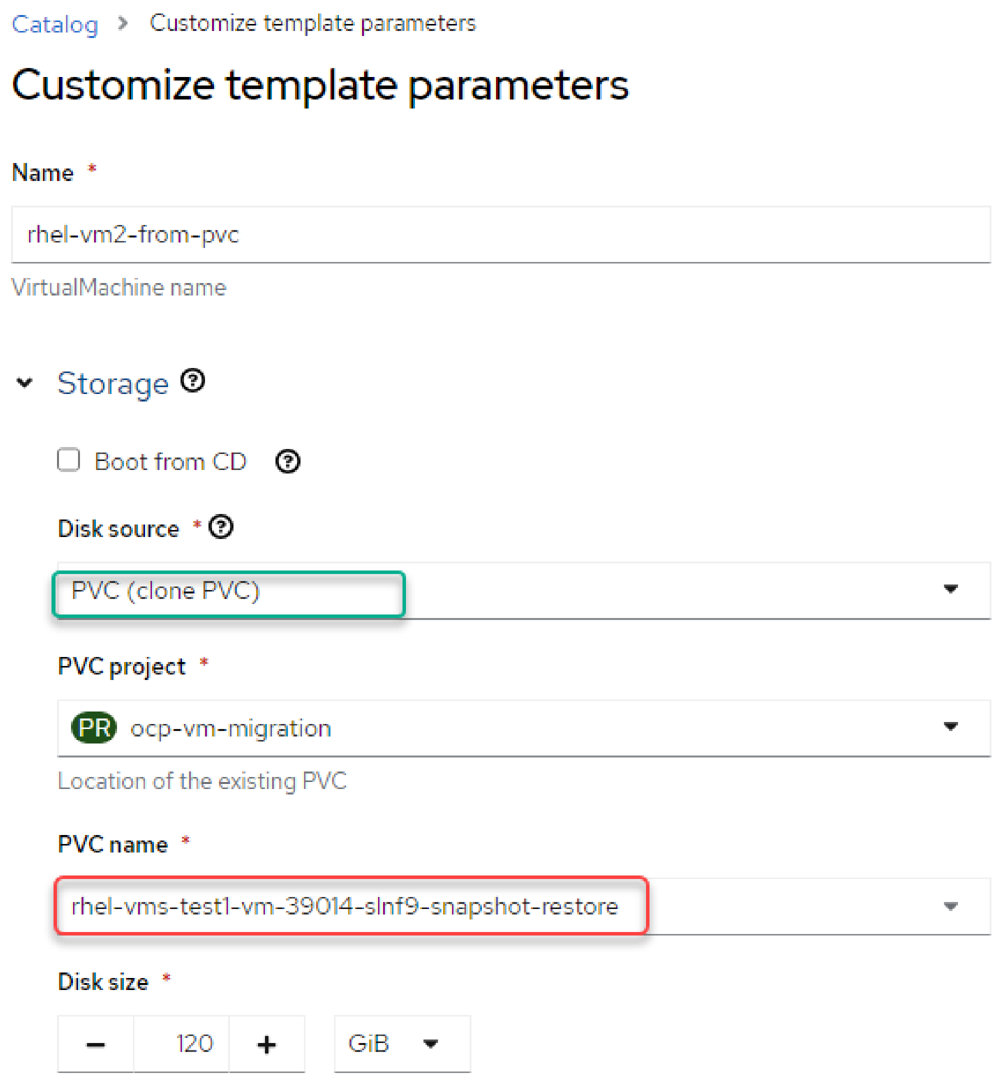
```
oc apply -f vspone-e1090-volumesnapshotclass.yaml
```



The next step is to restore the snapshot as a new PVC. This can be done from `Storage > VolumeSnapshots`. Select the recently created VolumeSnapshot, and click the ellipsis next to the snapshot and select `Restore as new PVC`.

Finally, you can create a new VM using this new PVC that was restored from the snapshot. This can be done from the command line or directly from the web console.

To do this from the web console, navigate to `Virtualization > VirtualMachines`, then click `Create`, select `From Template`, and then select a template without a bootable boot source. On the `Customize template parameters` page, expand `Storage` and select `PVC (clone PVC)` from the `Disk source` list. Then select the project and the PVC that were restored in the previous step. Make sure to set the disk size and click `Next`, then click `Create VirtualMachine`, as shown.



Create a VM by cloning PVCs

This process is very similar to creating a VM from a snapshot, except that you do not need to create a snapshot and then restore it to a PVC. Instead, you just need to create a clone of an existing PVC (from another VM), and then create a new VM. When creating the new VM, select `From Template` (template without a bootable boot source). On the Customize template parameters page, expand `Storage`, select `PVC (clone PVC)`, and select the cloned PVC. Then click `Create VirtualMachine`.

Connect a VM to a Linux bridge or services

In OpenShift Virtualization each VM is connected by default to the default internal pod network. To expose a virtual machine within the cluster or outside the cluster create a service object.

Another option is to either configure VM secondary network interfaces or change the current interface to a Linux bridge network, SR-IOV network, or OVN-Kubernetes secondary network. A Linux bridge emulates a hardware bridge to provide layer-2 networking, and in a virtualized environment such as OpenShift Virtualization it can be used to integrate VMs to the same network as the hosts. See https://docs.redhat.com/en/documentation/openshift_container_platform/4.17/html/virtualization/networking for more details.

You can always modify, add, or remove network interfaces by editing the VM specification.

The following are two examples of how to expose a VM using a service and how to connect a VM to a Linux bridge network (layer-2) in the same way a VM is connected in a VMware environment using standard or distributed port group.

Use a service to expose a VM

This example exposes a VM called `rhel-vm2` using a service.

Procedure

1. Use the `oc get vmi` command to list the VMs.

```
[root@jputilitysrv1 ocpjpcl1]# oc get vmi -A
NAMESPACE          NAME           AGE     PHASE     IP             NODENAME
ocp-vm-migration   rhel-vm2       7d10h   Running   10.130.0.158   ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab
ocp-vms            rhel9-sample   24h     Running   10.131.0.136   ocpjpcl1-worker-1.ocpjpcl1.ocp.sce.lab
```

2. Create a service to expose the VM. This can be done using the `oc` command or using the `virtctl` CLI tool:

```
virtctl expose vm rhel-vm2 -n ocp-vm-migration --port=22 --name=rhel-vm2-ssh
      --type=NodePort
```

Note: `NodePort` is used as an example; however, the preferred method is `Load balancer`.

This command creates the following service:

```
[root@jputilitysrv1 ocpjpcl1]# oc get vmi -A
NAMESPACE          NAME           AGE     PHASE     IP             NODENAME
ocp-vm-migration   rhel-vm2       7d10h   Running   10.130.0.158   ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab
ocp-vms            rhel9-sample   24h     Running   10.131.0.136   ocpjpcl1-worker-1.ocpjpcl1.ocp.sce.lab
```

3. Use SSH to access the VM using the worker node where the VM is running and using the port from the service.

```
[root@jputilitysrv1 ocpjpcl1]# ssh root@ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab -p 30336
The authenticity of host '[ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab]:30336 ([10.76.47.228]:30336)' can't be established
ECDSA key fingerprint is SHA256:OSajIeyaA+ZacA1vky0lR+LkVqoc4LpHcmrT3fMP54Q.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab]:30336' (ECDSA) to the list of known hosts.
root@ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab's password:
Activate the web console with: systemctl enable --now cockpit.socket

Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Wed May 29 23:37:42 2024 from 100.64.0.4
[root@rhel-vm2 ~]# hostname
rhel-vm2
[root@rhel-vm2 ~]#
```

Connect a VM to a Linux bridge network

The following example shows how to connect a VM to a Linux bridge network, and the example uses the same RHEL VM created previously.

Use this procedure to create a Linux bridge network and attach a VM to the network.

Procedure

1. Create a Linux bridge node network configuration policy (**NNCP**).
2. Create a Linux bridge network attachment definition (**NAD**) by using the web console or the command line.
3. Configure the VM to recognize the NAD by using the web console or the command line.

Create a Linux bridge node network configuration policy (NNCP)

Create the NNCP policy directly from the web console or from the CLI. This example uses the CLI.

Procedure

1. Install the Kubernetes NMState Operator. See the following links for more details:

   https://docs.redhat.com/en/documentation/openshift_container_platform/4.15/html/networking/kubernetes-nmstate#k8s-nmstate-about-the-k8s-nmstate-operator

   https://docs.redhat.com/en/documentation/openshift_container_platform/4.17/html/networking/kubernetes-nmstate#k8s-nmstate-about-the-k8s-nmstate-operator

2. Create the `NodeNetworkConfigurationPolicy` manifest.

```
apiVersion: nmstate.io/v1
kind: NodeNetworkConfigurationPolicy
metadata:
  name: br260-ens3f1-policy
spec:
  nodeSelector:
    node-role.kubernetes.io/worker: ""
  desiredState:
    interfaces:
      - name: br260
        description: Linux bridge with ens3f1 as a port
        type: linux-bridge
        state: up
        ipv4:
          enabled: false
        bridge:
          options:
            stp:
              enabled: false
          port:
            - name: ens3f1
```

   This example creates a policy `br260-ens3f1-policy` for a bridge interface type on the node's NIC `ens3f1`, but only on the worker nodes.

3. Use the `oc apply` command to create the NNCP.

```
oc apply -f NodeNetworkConfigurationPolicy.yaml

oc get nncp
```

```
NAME                  STATUS      REASON
br260-ens3f1-policy   Available   SuccessfullyConfigured
```

Result

After the status of the NNCP shows Available, you can use the CLI or GUI to verify in each of the worker nodes that a new bridge called `br260` has been created.

Example

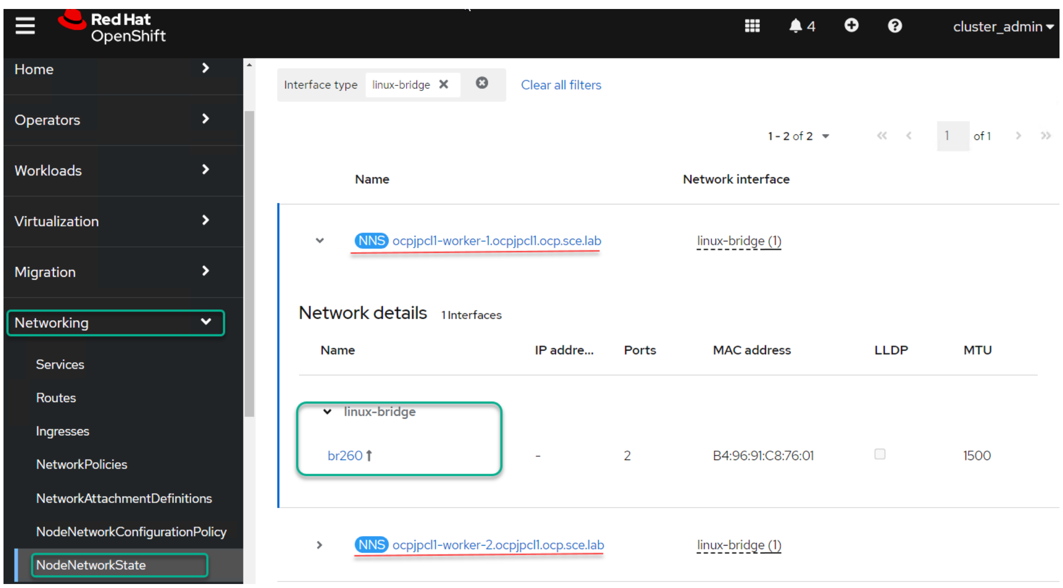
From the CLI, here is an example for `worker-1`:

```
[root@jputilitysrv1 ocpjpcl1]# oc debug node/ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab -
- chroot /host bash -c "ip a | grep br260"
Starting pod/ocpjpcl1-worker-2ocpjpcl1ocpscelab-debug-jmrx7 ...

5: ens3f1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq master br260 state UP
group default qlen 1000
755: br260: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group
default qlen 1000
```

Example

From the GUI, navigate to `Networking > NodeNetworkState`, expand one of the worker nodes, expand the network details, and a Linux bridge interface called `br260` is shown:



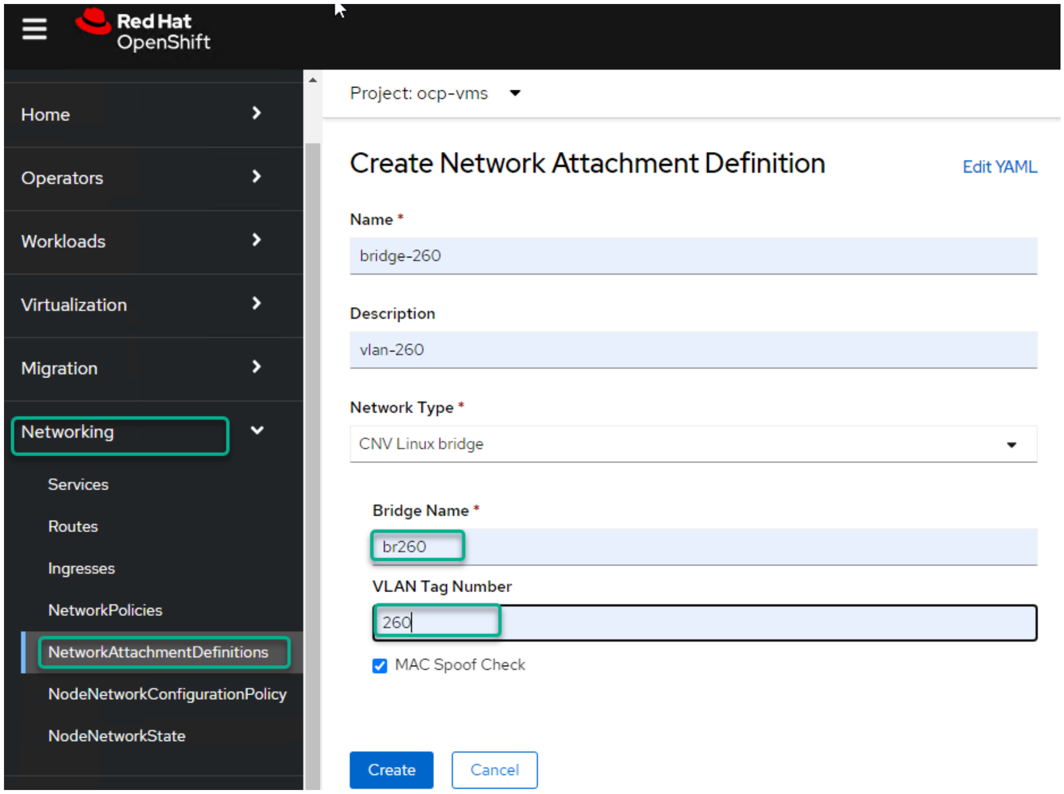
Create a Linux bridge network attachment definition (NAD)

The Linux bridge NAD to provide layer-2 networking to virtual machines can be created either from the web console or CLI. Use this procedure to create a Linux bridge NAD from the web console.

Procedure

1. In the web console, navigate to **Networking** > **NetworkAttachmentDefinitions**.

2. Click **Create Network Attachment definition** and enter the values as shown. Make sure the bridge name matches the bridge name used on the NNCP.



3. Click **Create**.

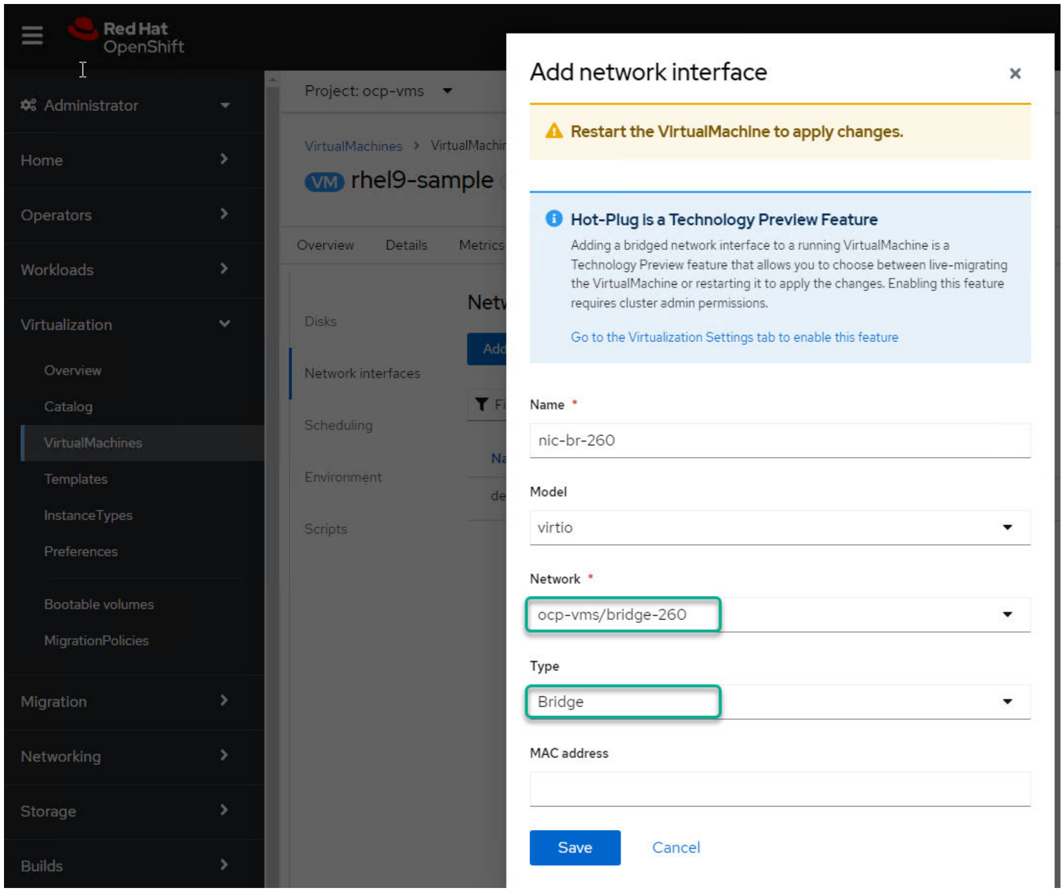
Configure the VM to recognize the NAD

The last step, after the NNCP and NAD have been created, is to add or modify a network interface to the VM.

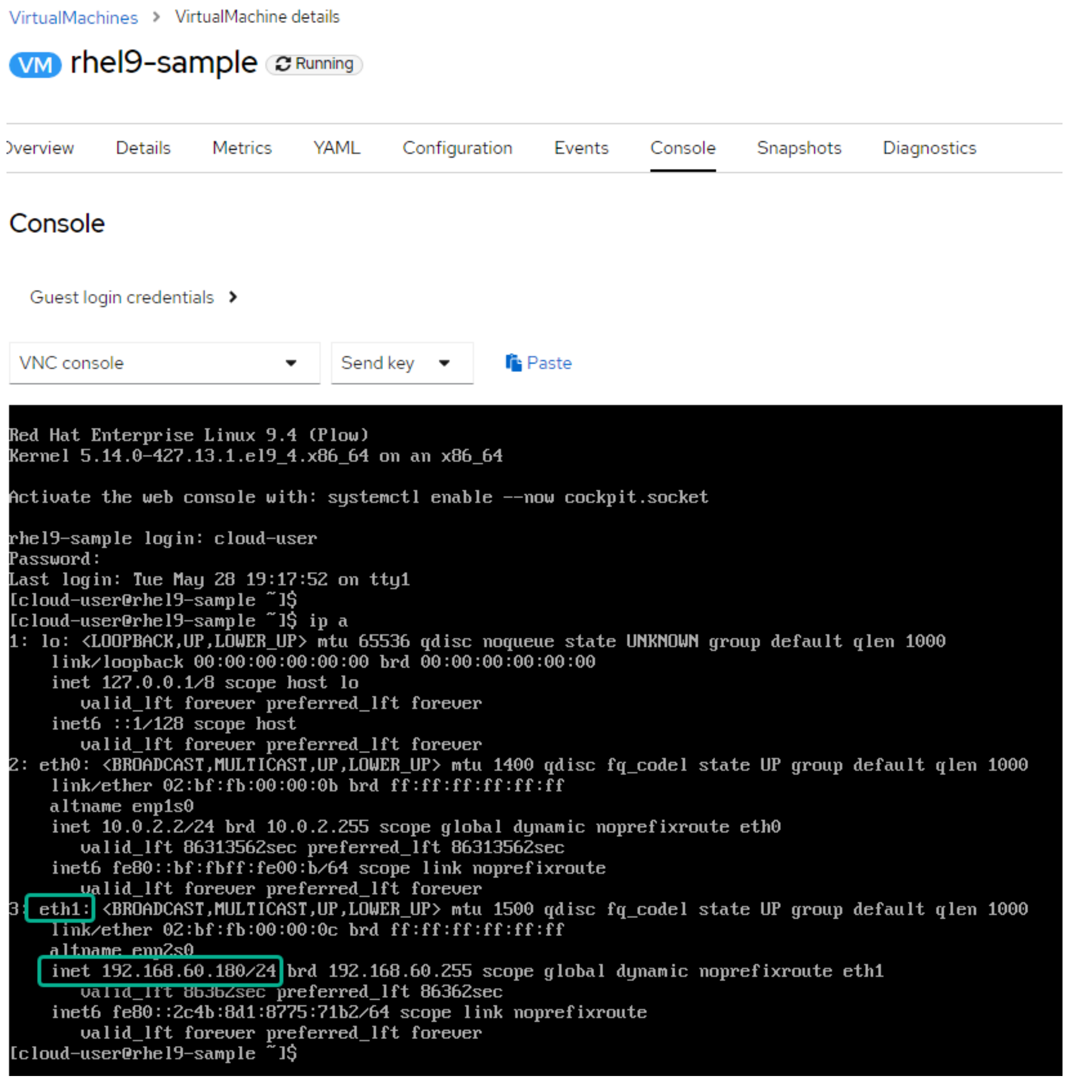
This example adds a new interface to the RHEL VM created previously. This procedure can be done from the web console or the CLI. This example uses the web console.

Procedure

1. On the web console, navigate to **Virtualization** > **VirtualMachines**.
2. Select the RHEL VM, select the **Configuration** tab, and then click **Network**.
3. Click **Add network interface** and make sure to select the network (NAD) previously created `bridge-260` and select **Bridge** for the type.



4. Save and restart the VM.
5. After the VM is restarted, log in to the VM and verify the presence of a new NIC, and if DHCP is enabled there should be an IP address already configured as shown.



Result

The procedure to connect a VM to a Linux bridge network is complete.

Live migration of virtual machines between nodes

Live migration is a "non-disruptive" VM migration. It is the process of moving a running VM instance from one node to another node in the OCP cluster without downtime. Note that this is different than a non-disruptive VM storage migration.

Live migration has the following requirements:

- The OCP cluster must have shared storage with `ReadWriteMany` (RWX) access mode. An OCP cluster backed by Hitachi Virtual Storage Platform and HSPC CSI driver already supports RWX access mode for block. Any VM created with a `storageClass` provisioned by Hitachi HSPC already uses RWX PVCs and can be live migrated without downtime.
- The cluster must have enough memory RAM and network bandwidth.

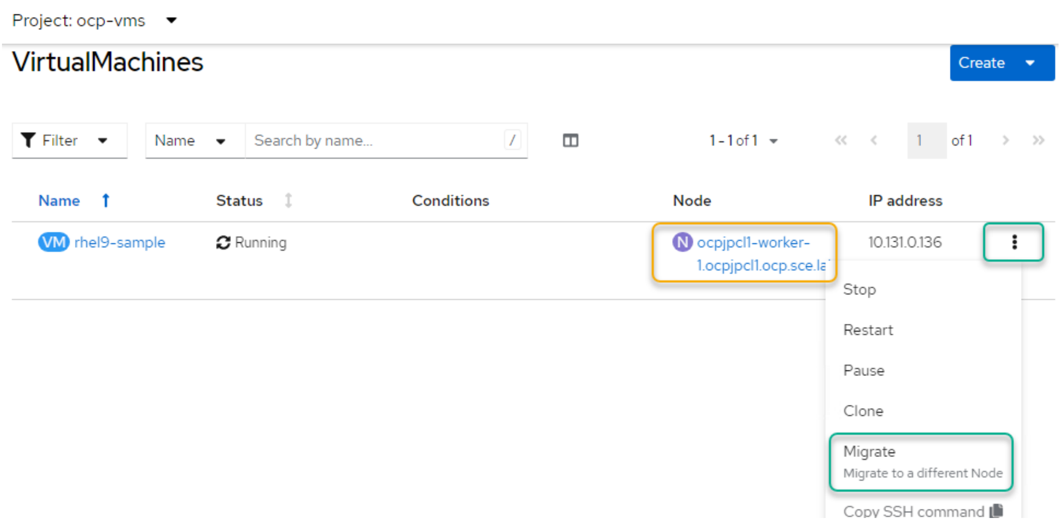
- The default number of migrations that can run in parallel in the cluster is 5, with a maximum of two (2) outbound migrations per node.
- Configuring a dedicated network for live migration is highly recommended.

Live migration policies can be used to apply different migration configurations to different groups of VMs, using any combination of labels such size, OS, and GPU.

A live migration can be triggered from the GUI, CLI, API, or automatically. The following example shows the live migration of an RHEL VM from `worker-1` to `worker-2` initiated from the web console.

Procedure

1. In the web console, navigate to **`Virtualization`** > **`VirtualMachines`**.
2. From the VM you want to migrate, click the ellipsis (three dots), then click **`Migrate (Migrate to a different node)`**.



3. You can verify the status of the migration by refreshing the web console or using the following command:

   ```
   oc describe vmi <vm_name> -n <namespace>
   ```

   The following example shows the command output, showing the status of the migration operation, the source, target node, and start/end time stamps. This migration was completed successfully without any downtime.

```
[root@jputilitysrv1 ocpjpcl1]# oc describe vmi rhel9-sample -n ocp-vms

...
Status:
  Conditions:
    Type:                  Ready
    Last Transition Time:  <nil>
    Status:                True
    Type:                  LiveMigratable
  Guest OS Info:
    Id:              rhel
    Kernel Release:  5.14.0-427.13.1.el9_4.x86_64
  Interfaces:
...
    Info Source:     domain, guest-agent, multus-status
    Interface Name:  eth1
    Ip Address:      192.168.60.180
    Mac:                            02:bf:fb:00:00:0c
    Name:                           nic-br-260
  Migration Method:  BlockMigration
  Migration State:
    Completed:       true
    End Timestamp:  2024-05-29T06:56:02Z
    Migration Configuration:
      Node Drain Taint Key:                     kubevirt.io/drain
      Parallel Migrations Per Cluster:          5
      Parallel Outbound Migrations Per Node:    2
    Mode:                                       PreCopy
    Source Node:                                ocpjpcl1-worker-1.ocpjpcl1.ocp.sce.lab
    Start Timestamp:                            2024-05-29T06:56:00Z
    Target Node:                        ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab
    Target Node Address:                10.130.0.40
    Target Node Domain Detected:        true
    Target Node Domain Ready Timestamp: 2024-05-29T06:56:02Z
    Target Pod:                         virt-launcher-rhel9-sample-2gpxt
  Migration Transport:                  Unix
  Node Name:                            ocpjpcl1-worker-2.ocpjpcl1.ocp.sce.lab
  Phase:                                Running
```

Migrate virtual machines from VMware into OpenShift Virtualization

Migration Toolkit for Virtualization provides an easy-to-use UI and allows for individual or mass migration of VMs from VMware vSphere, Red Hat Virtualization and OpenStack to OpenShift or between OpenShift clusters and integrated with VSP One using the HSPC CSI driver.

Prerequisites

Prerequisites vary depending on the source provider and the type of migration (cold or warm). In this paper the focus is on VMware as the source provider and OpenShift Virtualization as the destination provider:

- VMware vSphere cluster prerequisites:
  - Have a compatible version of VMware vSphere. Always check the software compatibility guidelines of MTV.
  - Migration Toolkit for Virtualization (MTV) uses the VMware Virtual Disk Development Kit (VDDK) SDK to accelerate transferring virtual disks from VMware vSphere. Creating a VDDK image, although optional, is highly recommended.
  - Make sure the virtual machine guest OS is certified and supported with OpenShift Virtualization.
  - VMware Tools is required on the source virtual machine only if you are using a pre-migration hook that requires access to the virtual machine.
  - For warm migration, you must enable change block tracking (CBT) on the VMs and on the VM disks.
- OpenShift cluster prerequisites:
  - Because MTV is an add-on to the OCP cluster, before installing MTV make sure the OCP cluster meets all the prerequisites described in Red Hat OpenShift Virtualization with Hitachi VSP One storage (on page 18).
- Network prerequisites:
  - IP addresses, VLANs, and other network configuration settings must not be changed before or during migration. The MAC addresses of the virtual machines are preserved during migration.
  - The network connections between the source environment, the OpenShift Virtualization cluster, and the replication repository must be reliable and uninterrupted.
  - If you are mapping more than one source and destination network, you must create a network attachment definition for each additional destination network.

See Migration Toolkit for Virtualization at https://access.redhat.com/documentation/en-us/migration_toolkit_for_virtualization for more details.

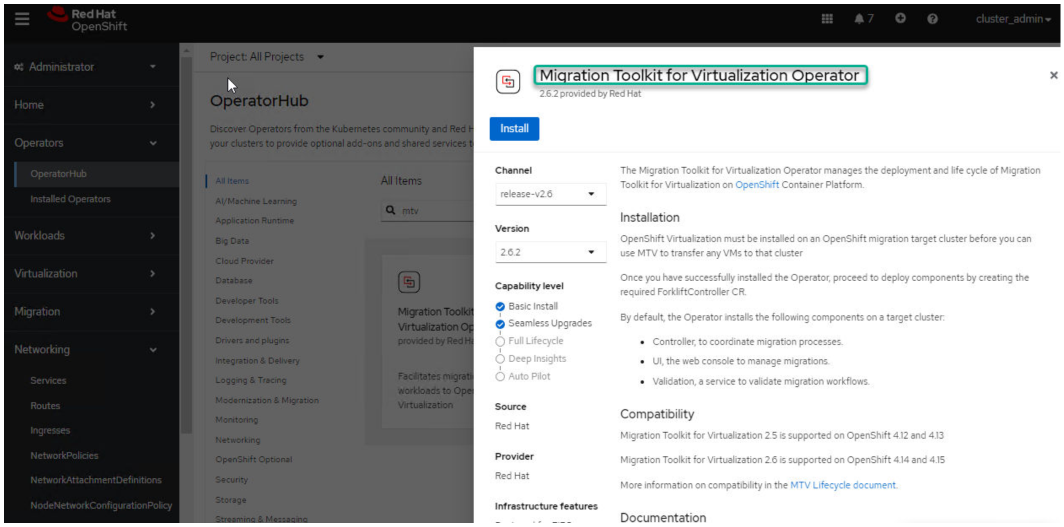
Install and configure the Migration Toolkit for Virtualization (MTV)

The Migration Toolkit for Virtualization (MTV) Operator can be deployed from the web console or the CLI. This operator includes the Migration plugin for the OCP web console. In this paper we are using the web console and the OperatorHub.

Complete this procedure to deploy the MTV Operator using the web console.

Procedure

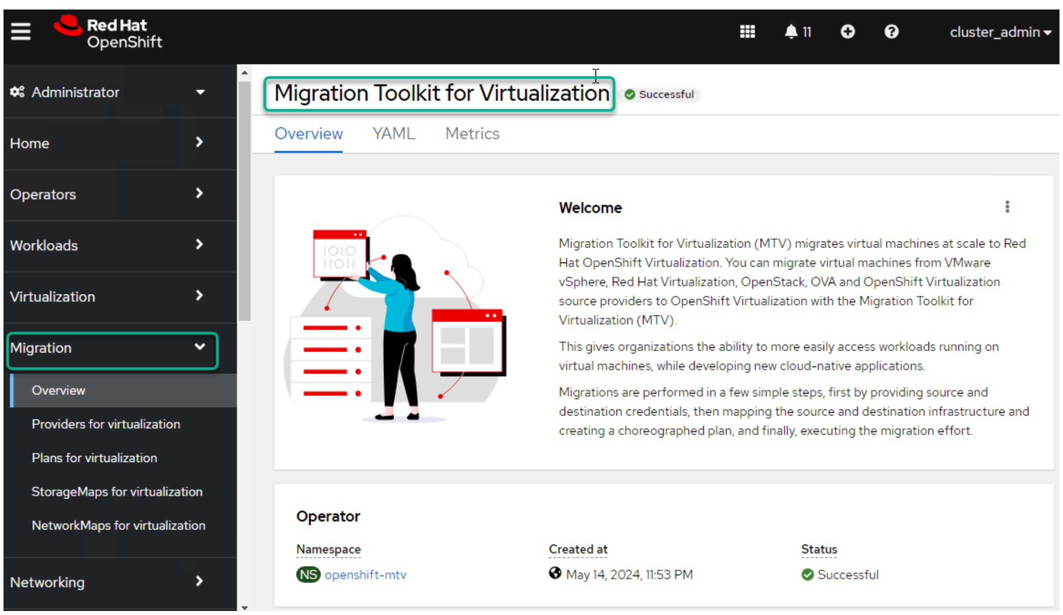
1. Log in to the Red Hat OCP web console as a user with cluster-admin permissions.
2. Navigate to **Operators** > **OperatorHub** and filter by the keyword `mtv`.
3. Select **Migration Toolkit for Virtualization Operator** and click **Install**.



4. Wait for the operator installation to complete, click **Create ForkliftController**, and click **Create**.
5. (Optional): Use the OCP web console or CLI to verify that the MTV pods are running.

Next steps

When the plugin is ready, reload the page and you will see a new option called Migration on the OCP web console. The following illustration shows the MTV user interface integrated into the OCP web console.

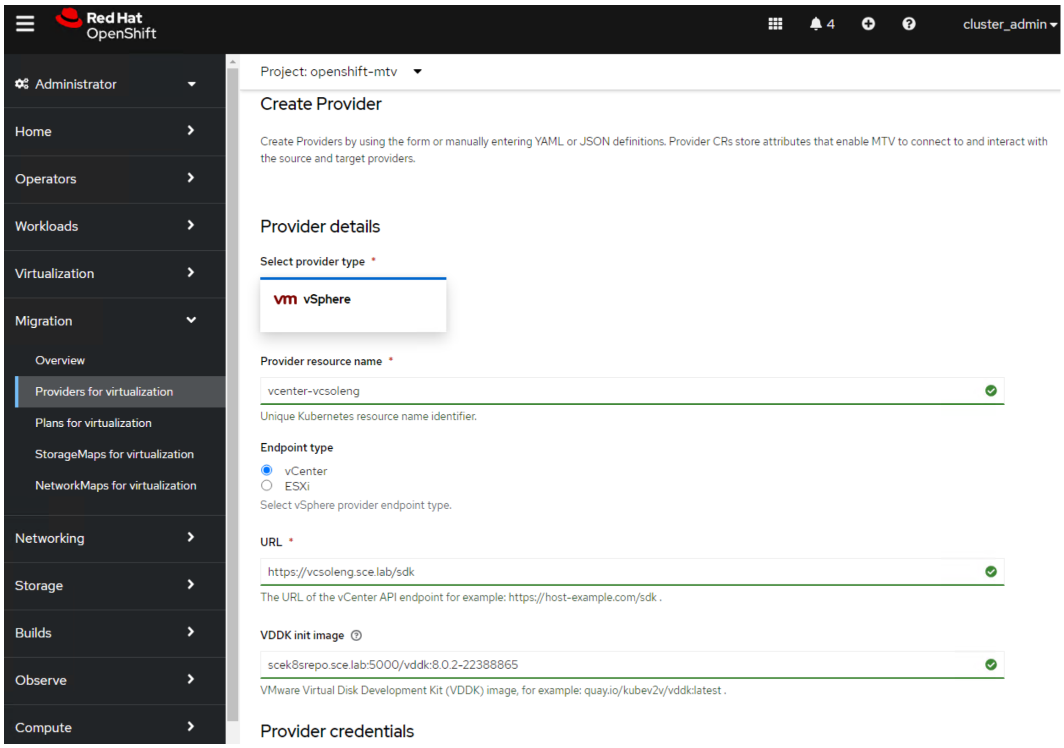


VM migration procedure

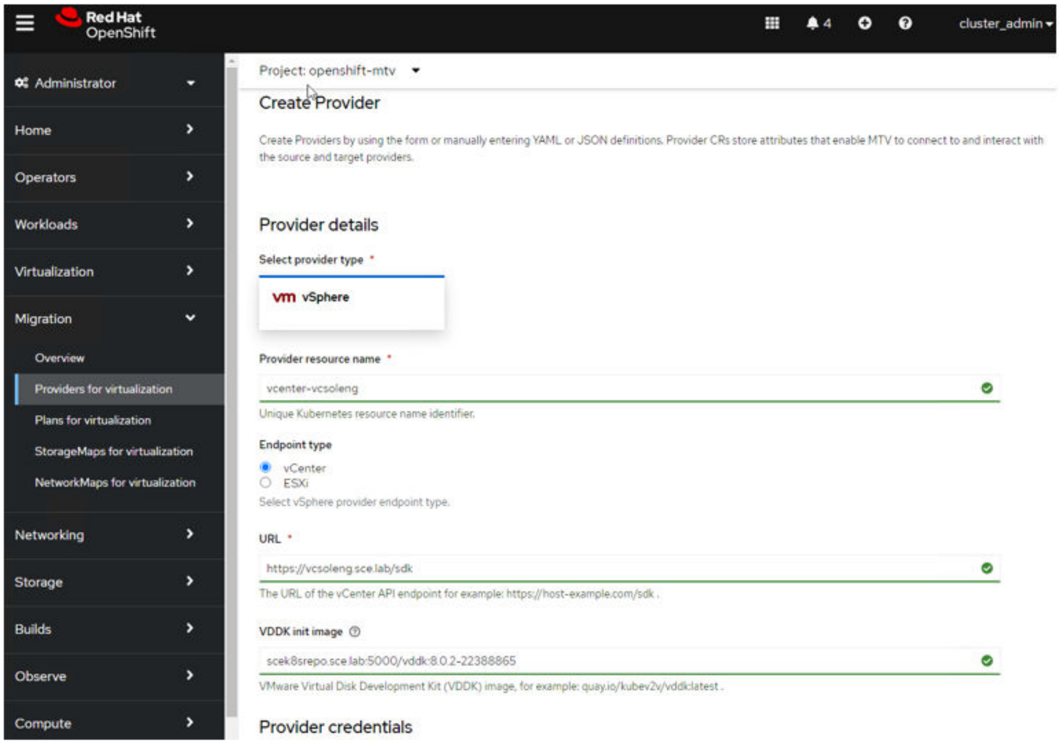
Migrations are performed in a few simple steps, first by providing a source provider, then creating a migration plan which includes mapping the source and destination infrastructure (storage and network mappings), and finally, running the migration.

Procedure

1. Add source providers for the migration.
    a. In the web console, navigate to **Migration** > **Providers for Virtualization**.
    b. Select a project (namespace), click **Create Provider**, and then choose **vSphere**.
    c. Provide all the details for the VMware vCenter environment and VDDK init image, and then click **Create provider**.



> **Note:** MTV uses the VMware Virtual Disk Development Kit (VDDK) SDK to accelerate transferring virtual disks from VMware vSphere. Creating a VDDK image, although optional, is highly recommended. It is also important to have an image registry to host the VDDK image, and make sure to use the appropriate version of VDDK according to your vSphere environment. Follow the instructions provided at https://docs.redhat.com/en/documentation/migration_toolkit_for_virtualization/2.8/html/installing_and_using_the_migration_toolkit_for_virtualization/prerequisites_mtv#network-prerequisites_mtv to download, build a VDDK VDDM container image, and push the VDDK image to your image registry. The image registry must be accessible from the OpenShift cluster.



   d. When the provider is in **Ready** state you can start creating a migration plan for the next steps.

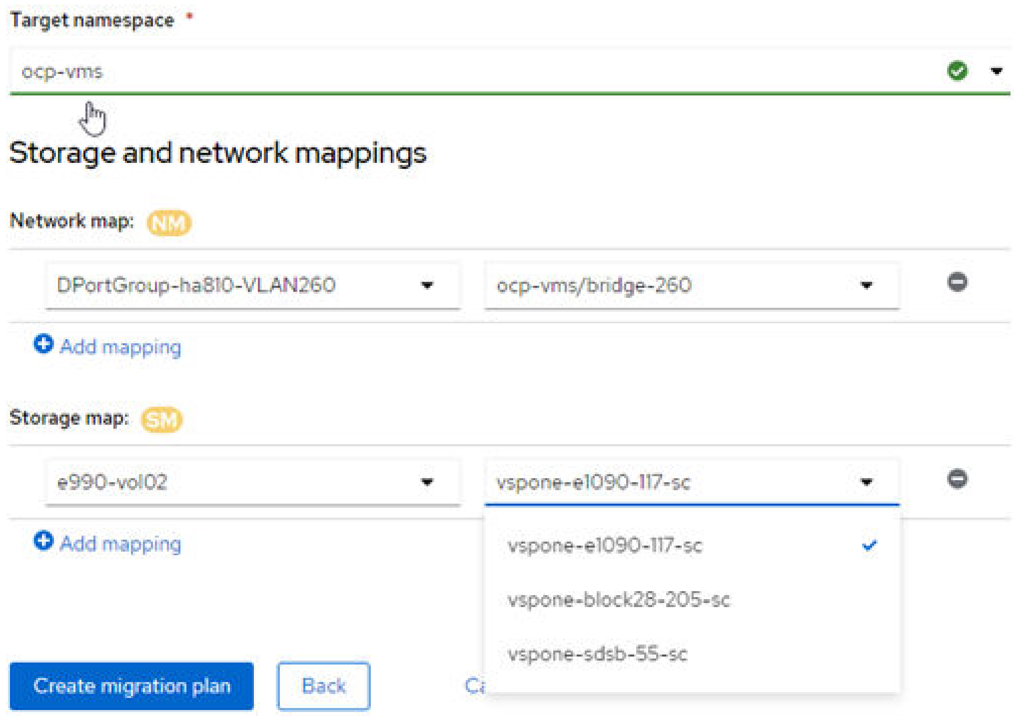
2. Create a migration plan and select VMs from the source provider for migration.

   While there is a need to create StorageMaps and NetworkMaps for virtualization, this can be accomplished as part of the workflow of the migration plan.

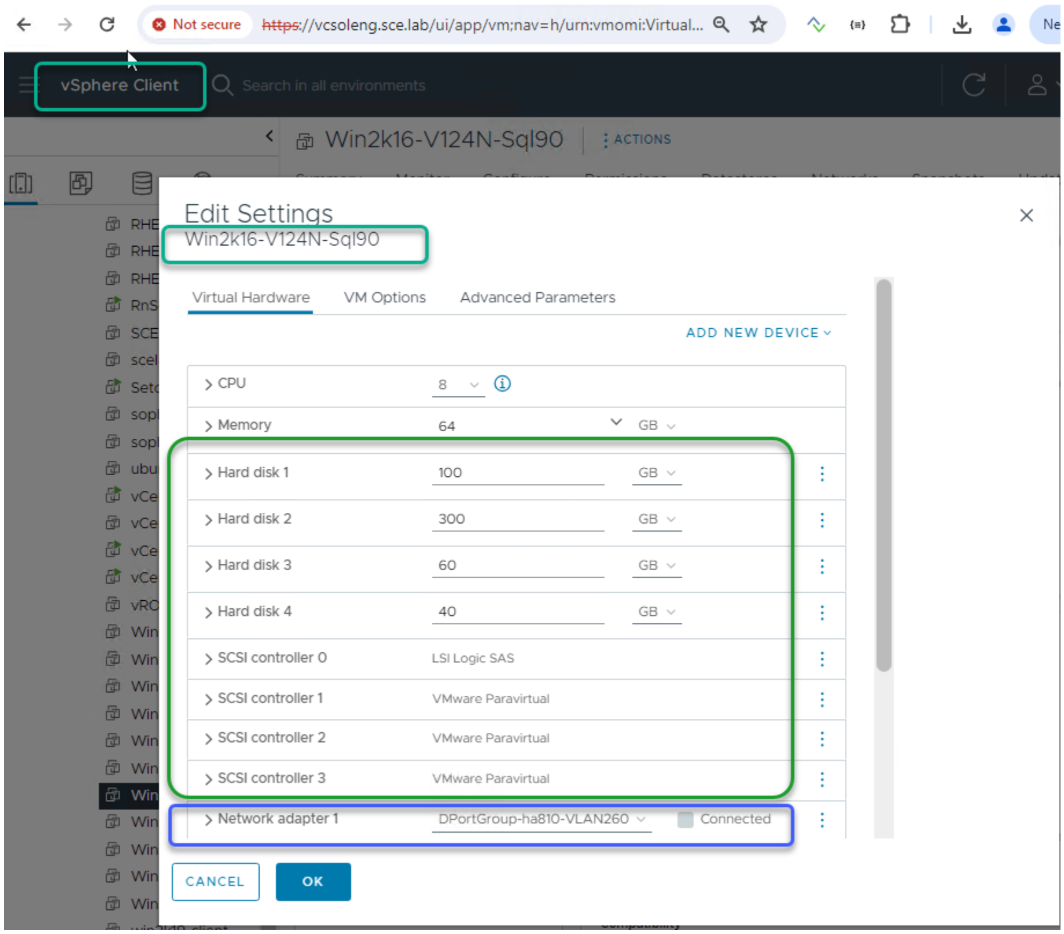
   a. Click **Plans for virtualization**, and then click **Create plan**.

   b. Select the source provider created previously called `vcenter-vcsoleng`.

   c. Select the VMs that you want to migrate to OpenShift Virtualization.

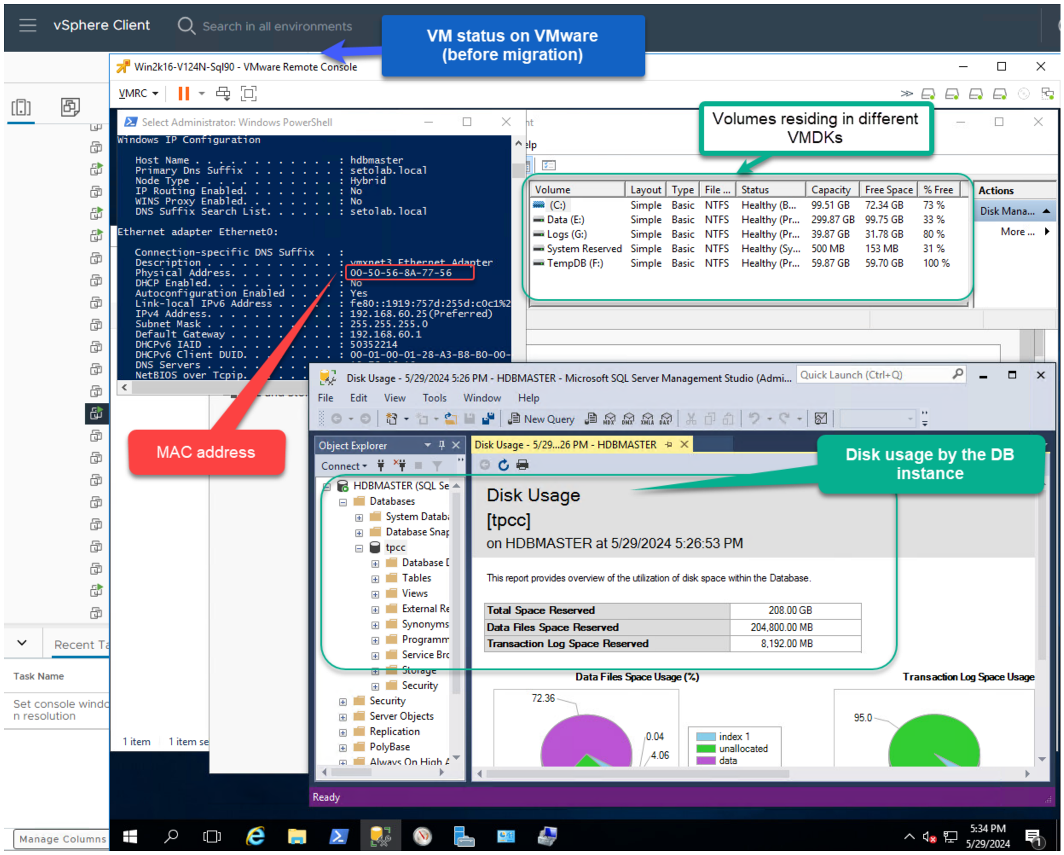


This diagram shows the status of the VM to be migrated as seen on VMware vCenter UI, which is a Windows VM with SQL Server installed, using 4 virtual disks (VMDKs) in a VMFS datastore, with each disk using a separate SCSI controller, and connected to a DVPortGroup using VLAN 260.



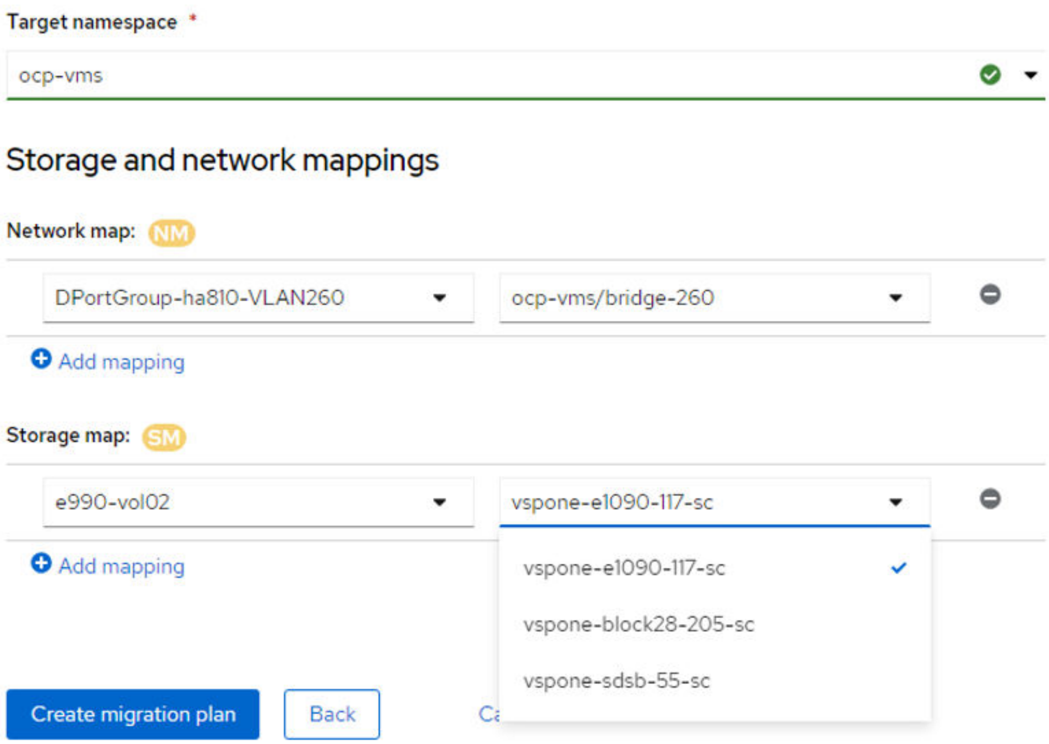
The following illustration shows the status of the Windows Server VM with a running instance of SQL Server using multiple disks for OS, DB data, logs, and temp partitions on the VMware environment. This example is for a cold migration, which is why the VM is powered off before starting the migration.

Note: While the example is for cold migration, OpenShift Virtualization supports warm migrations which require a minimal maintenance window.



d. Click **Next** and then enter a name for the migration plan. Change the target namespace. In this case `ocp-vms`.

   This is the step where you need to select the storage and network mappings. For the network map, select the Linux bridge network. And for storage, select one of the Storage Classes defined for VSP storage.

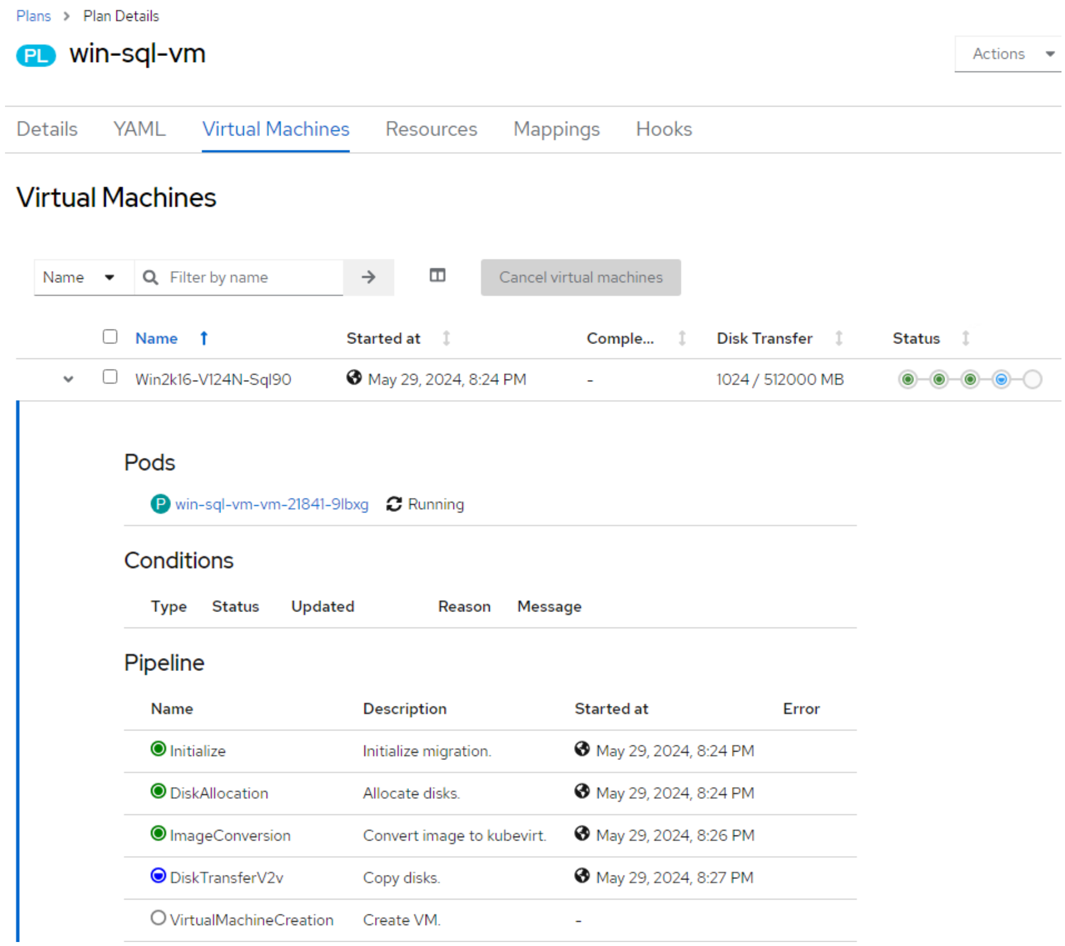


e. Click **Create migration plan** and wait until the plan details show **Start migration** in green.

3. Run the migration plan.

   a. Click **Start migration** to start the process and click **Start**.

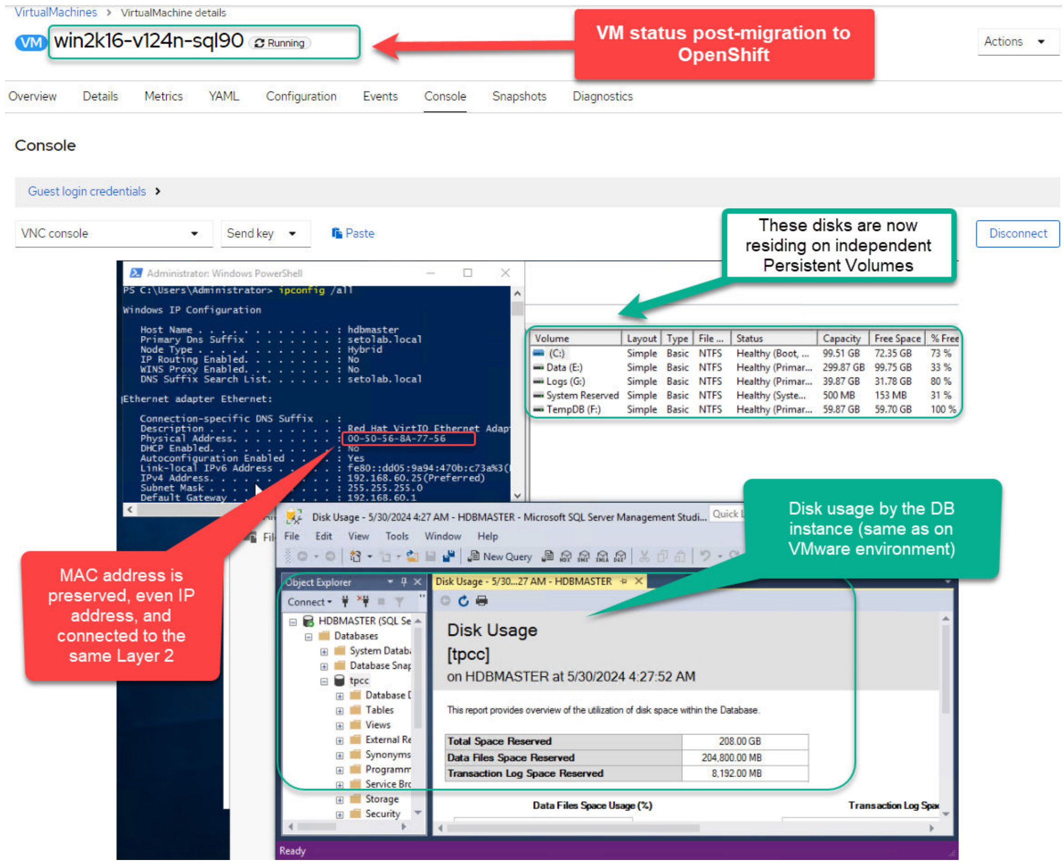
   b. To see details of the migration, select the **Virtual Machines** tab, and then expand details for the VM to see details for the pipeline migration.



c. After the migration is complete, you can see more details for the configuration of the VM either from the GUI or CLI. Here is an example of the four PVCs, one for each of the original virtual disks.

```
[root@jputilitysrv1 ocpjpcl1]# oc get pvc
NAME                          STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS
rhel9-sample                  Bound    pvc-0d12b33c-5ce6-4fbf-9a26-ed6a1e046917   30Gi       RWX            vspone-e1090-117-sc
win-sql-vm-vm-21841-21mzd     Bound    pvc-e6c2c624-f3b1-48ad-99cf-f8721f8f6fe9   40Gi       RWX            vspone-e1090-117-sc
win-sql-vm-vm-21841-48cjg     Bound    pvc-f9251cf2-1db6-4e85-8c54-300ba87cd74d   100Gi      RWX            vspone-e1090-117-sc
win-sql-vm-vm-21841-chk6h     Bound    pvc-28bfb3b0-29ec-4edc-90a1-2759cef89ba8   60Gi       RWX            vspone-e1090-117-sc
win-sql-vm-vm-21841-q9k7p     Bound    pvc-f873ad84-9fd8-499b-a160-80b192393a9e   300Gi      RWX            vspone-e1090-117-sc
[root@jputilitysrv1 ocpjpcl1]#
```

d. After the VM is started on the OpenShift Virtualization, confirm that the VM maintains the same MAC address, the used disk is the same as the one before the migration, and the application (SQL Server) is running. You can also access the VM in the same way as the original VM. While the following illustration shows the console from OpenShift, it is possible to connect to the VM using RDP as well.



Migrate virtual machines with storage offload

If you are migrating VMs from vSphere clusters to OpenShift Virtualization and both are backed by VSP One storage, you can take advantage of the storage offload feature. This feature was introduced in Migration Toolkit for Virtualization (MTV) Operator 2.9.

Make sure you have the latest version installed before proceeding. As of MTV 2.9, only cold VM migration is supported. Verify that the latest MTV version supports warm VM migration if it is required in your environment.

The following configuration requirements must be met before proceeding:

- The Fibre Channel or iSCSI storage protocols are used for vSphere and OpenShift.
- In VSP storage, set Host Mode Option (HMO) 54 and 63 to enable the VMware VAAI feature for vSphere environments.

Note: NVMe-over-fabric and TCP NVMe are not supported.

As of MTV Operator 2.9.2, you must install the vmkfstools VIB package to each ESXi node. This requirement might not be needed in the latest MTV version, so see the latest release notes. The vmkfstools package is used to issue the XCOPY storage offload command from the ESXi host, and you might need to set this before installing the VIB:

```
esxcli software acceptance set --level=CommunitySupported
```

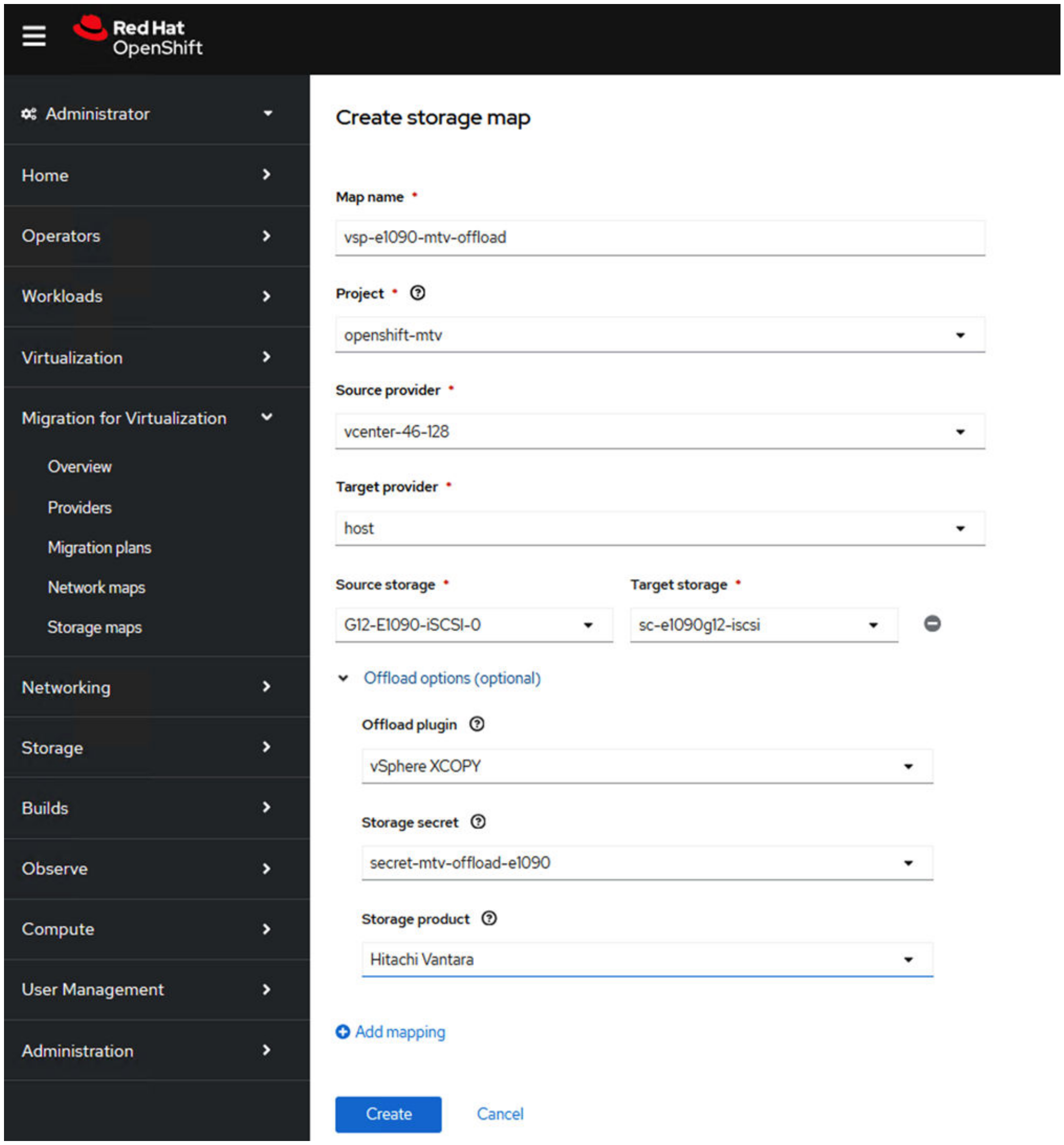
Follow this link to install the VIB:

https://github.com/kubev2v/forklift/tree/main/cmd/vsphere-xcopy-volume-populator/vmkfstools-wrapper

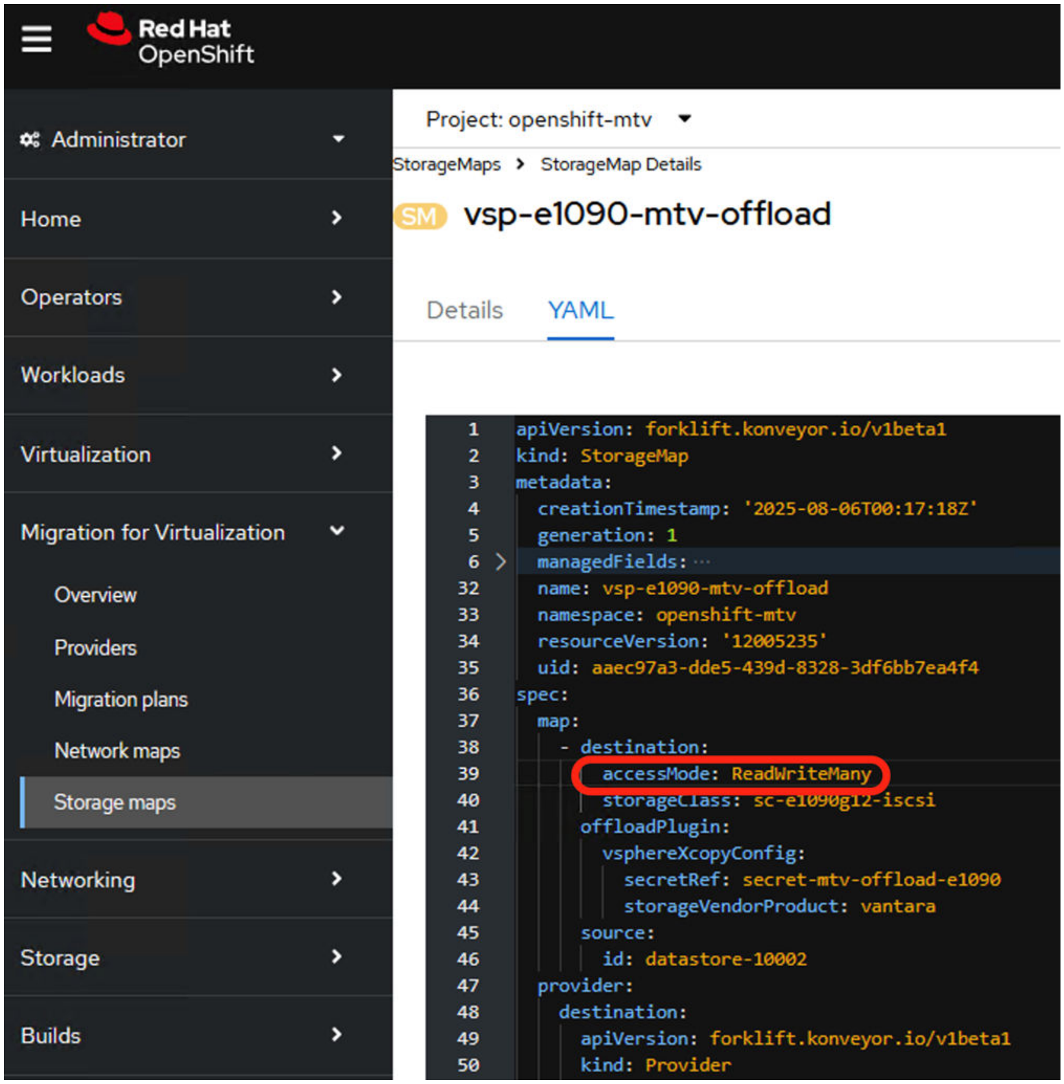
1. Create a VSP One storage access secret with the following example. This secret will be used by MTV to access VSP One storage.

```
kind: Secret
apiVersion: v1
metadata:
  name: secret-mtv-offload-e1090
  namespace: openshift-mtv
data:
  # base64 encoded storage hostname URL. E.g.: echo -n "https://172.25.11.111" |
base64
  STORAGE_HOSTNAME: aHR0cHM6Ly8xNzIuMjUuNDQuMTE2
  # base64 encoded storage serial number. E.g.: echo -n "123456" | base64
  STORAGE_ID: MTIzNDU2
  # base64 encoded storage host URL. E.g.: echo -n "https://172.25.11.111" |
base64
  STORAGE_URL: aHR0cHM6Ly8xNzIuMjUuNDQuMTE2
  # base64 encoded storage port 443. E.g.: echo -n "443" | base64
  STORAGE_PORT: NDQz
  # base64 encoded ESXi Host Groups with ":" separator.
  # E.g.: echo -n "CL2-C,3:CL3-C,3" | base64
  HOSTGROUP_ID_LIST: Q0wyLUMsMzpDTDMtQywz
  # base64 encoded storage user name. E.g.: echo -n "user1" | base64
  STORAGE_USERNAME: dXNlcjE=
  # base64 encoded storage password. E.g.: echo -n "Password1" | base64
  STORAGE_PASSWORD: UGFzc3dvcmQx
type: Opaque
```

2. Create a StorageMap with the offload feature in Migration for Virtualization > Storage maps and click Create storage map > Create with form. Make sure to set Offload options as shown.



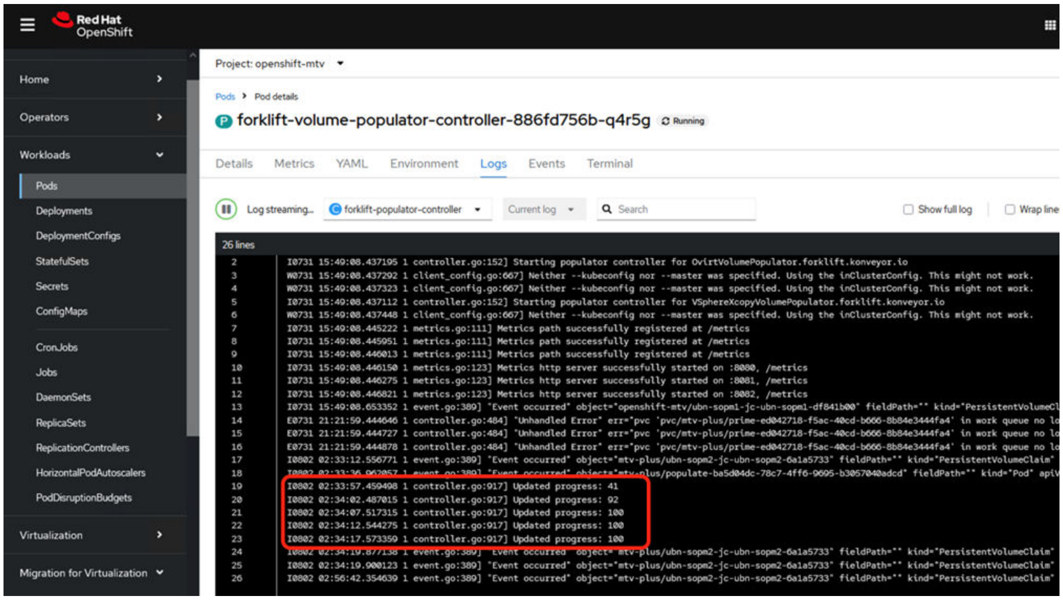
3. After the StorageMap is created, click the YAML tab to see if accessMode: ReadWriteMany exists. If it is missing, add it as shown.



4. Create a MigrationPlan and use the StorageMap with storage offload options that you just created. Start the VM migration after it is in the ready state. Refer to the previous section if needed.

Check the storage offload

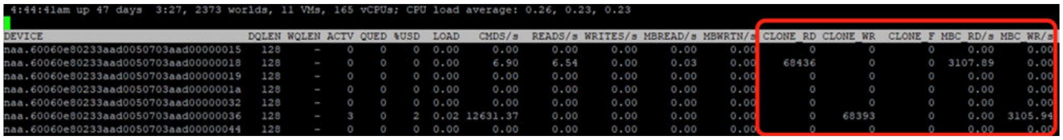
During the VM migration, you can look for the following `forklift-volume-populator-controller` pod and monitor the log file. This pod is created only when the storage offload is running correctly.



You can also run `esxtop` to monitor the storage offload VM migration from an ESXi host in real time.

- SSH into the ESXi that hosts the target VMs.
- Run the `esxtop` command, and press “u” to monitor the attached disks.
- Then press “f” to see the following options.
- Select the “O” option and optionally “P” and press Enter.

When the storage offload is happening, you will see a number of commands by issuing `CLONE_RD` and `CLONE_WR` as shown. This means VM data copying is offloaded within the VSP storage.



4:44:41am up 47 days 3:27, 2373 worlds, 11 VMs, 165 vCPUs; CPU load average: 0.26, 0.23, 0.23

DEVICE	DQLEN	WQLEN	ACTV	QUED	%USD	LOAD	CMDS/s	READS/s	WRITES/s	MBREAD/s	MBWRTN/s	CLONE_RD	CLONE_WR	CLONE_F	MBC_RD/s	MBC_WR/s
naa.60060e80233aad0050703aad00000015	128	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0.00	0.00
naa.60060e80233aad0050703aad00000018	128	-	0	0	0	0.00	6.90	6.54	0.00	0.03	0.00	68436	0	0	3107.89	0.00
naa.60060e80233aad0050703aad00000019	128	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0.00	0.00
naa.60060e80233aad0050703aad0000001a	128	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0.00	0.00
naa.60060e80233aad0050703aad00000032	128	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0.00	0.00
naa.60060e80233aad0050703aad00000036	128	-	3	0	2	0.02	12631.37	0.00	0.00	0.00	0.00	0	68393	0	0.00	3105.94
naa.60060e80233aad0050703aad00000044	128	-	0	0	0	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0.00	0.00

Items to note post-migration of VMs

- For Linux VMs:
  - After Linux VMs are migrated to OpenShift Virtualization, pay attention to the network device naming policies. For example, an interface named `ens192` on the original VM will be renamed `enp1s0` after migration. If the interface is down, one option is to update the network settings and specify the correct device name, and then restart the network services.
- For Windows VMs:
  - Uninstall VMware tools after migration because the migrated VM uses QEMU/VirtIO drivers.
  - If the VM had a static IP, you might need to remove the ghost network interfaces using instructions from the KB article at https://knowledge.broadcom.com/external/article/343044/networking-error-ip-address-already-assi.html. A reboot might be needed.
  - If the VM had several disks (VMDKs), only the OS/boot disk will be online by default. Log in to the VM and use the Disk Management utility to bring the other disks online.

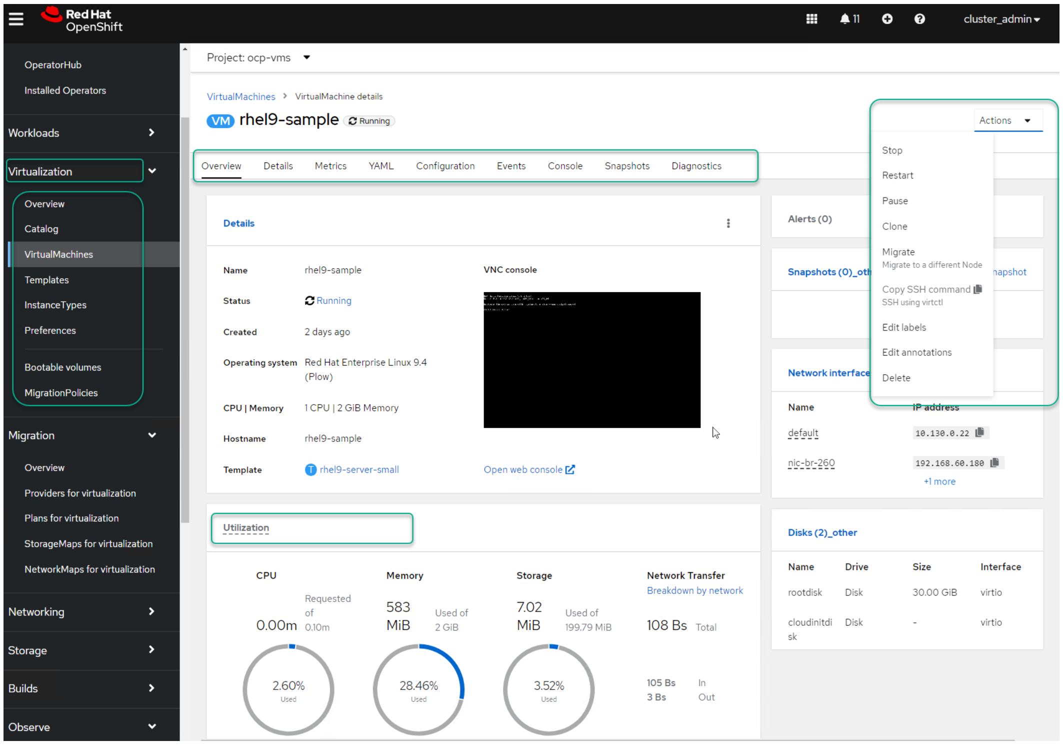
    This is something that might be fixed in future versions of the MTV operator. Also, the disks can be brought online using migration hooks as part of the migration plan on MTV
- This applies to both Windows and Linux VMs:
  - MTV automatically embeds the QEMU guest agent into the migrated VM. In Windows you can verify this either on the Programs and Features or from the command line using `net start` and verify that the output contains `QEMU Guest Agent`.
  - If needed you can always install or update the QEMU guest agent and VirtIO drivers following instructions from *Installing the QEMU guest agent and VirtIO drivers* at https://docs.redhat.com/en/documentation/openshift_container_platform/4.17/html/virtualization/virtual-machines#virt-uploading-image-virtctl_virt-creating-vms-uploading-images.

Manage and monitor virtual machines

Virtual machine instances (VMI) resources in an OpenShift Container Platform cluster can be managed either using the OpenShift web console or using the `oc` or `virtctl` commands from the command line interface (CLI).

The `virtcli` command provides more virtualization options than the `oc` command. For example, you can use `virtctl` to pause a VM or expose a port as seen in previous use cases. The `virtctl` utility is available for Linux, Windows or Mac, and can be downloaded directly from the OpenShift cluster web console, under `Virtualization > Overview` (Download virtctl).

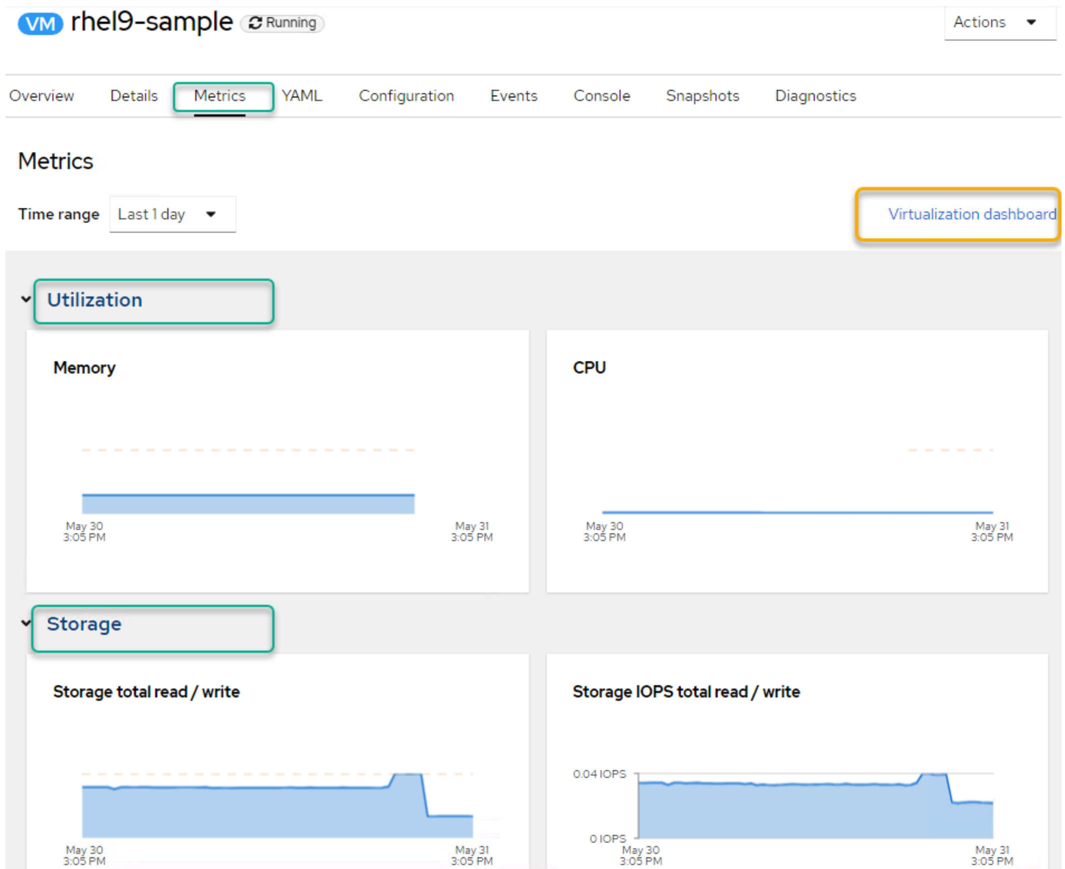
OpenShift Virtualization adds new objects into the OpenShift Container Platform cluster to enable virtualization tasks. With these new features you can create and manage VMs, connect to the VMs using the web console or CLI tools, import and clone existing VMs, and perform many other advanced tasks related to VM resources.



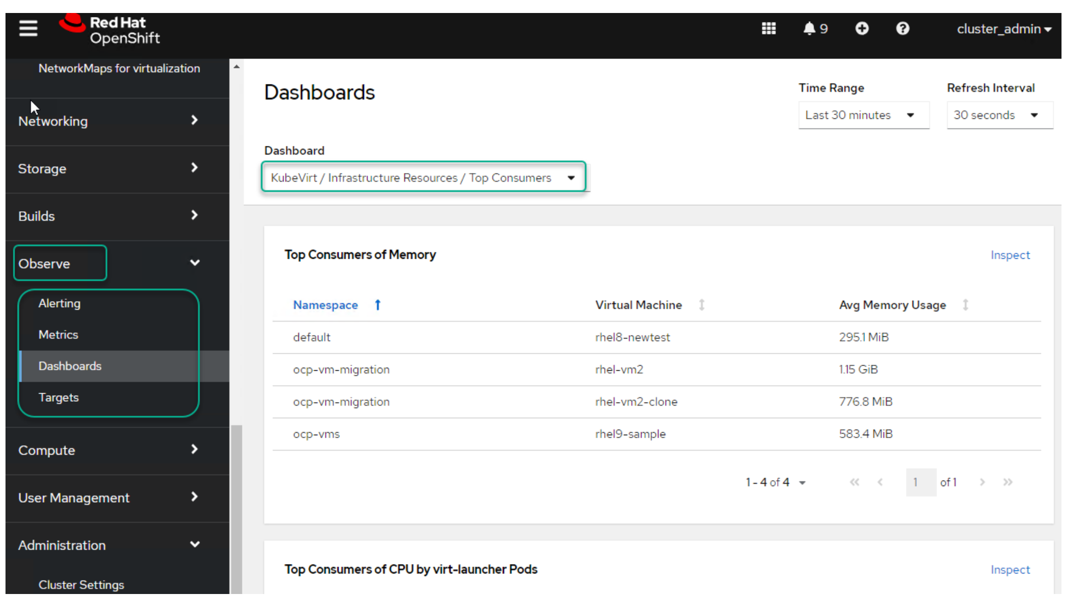
When it comes to monitoring, OpenShift Container Platform and OpenShift Virtualization provide many tools to help monitor the health of your cluster, virtual machines, and container applications. Among these tools are the following:

- Monitoring OpenShift Virtualization VM health status
- OpenShift Container Platform cluster checkup framework
- Prometheus queries for virtual machines
- VM custom metrics
- VM health status
- Runbooks, to diagnose and resolve issues triggered by OpenShift Virtualization

The following is an example of the metrics of one of the VMs. It shows detailed use of the virtual machine, and if you click any of the graphs you will be taken to the detailed metrics. The metrics under `Observer > Metrics` are collected by Prometheus, an OpenShift service for VMs. It even provides a `query` command that you can customize to create your own dashboards if needed.



There is also a dashboard for virtualized resources. To access it, either click the `Virtualization dashboard` directly from the VM's `Metrics` tab or navigate to `Observer > Dashboards`, and under `Dashboards` select `Kubevirt / Infrastructure Resources`. You can change the time range and refresh interval as shown.



See https://docs.redhat.com/en/documentation/openshift_container_platform/4.17/html/virtualization/monitoring#virt-monitoring-overview for more details.

Monitor Kubernetes resources and Hitachi storage with Hitachi Storage Plug-in for Prometheus

Hitachi Storage Plug-in for Prometheus enables the Kubernetes administrator to monitor the metrics of Kubernetes resources and Hitachi storage system resources within a single tool. Hitachi Storage Plug-in for Prometheus uses Prometheus to collect metrics and Grafana to visualize those metrics for easy evaluation by the Kubernetes administrator. Prometheus collects storage system metrics such as capacity, IOPS, and transfer rate in five-minute intervals.

For more details see https://docs.hitachivantara.com/search/all?query=Hitachi+Storage+Plug-in+for+Prometheus&content-lang=en-US and https://community.hitachivantara.com/blogs/jose-perez/2022/02/22/monitoring-kubernetes-resources-and-hitachi-storag.

Conclusion

Hitachi Virtual Storage Platform One Block, Hitachi Storage Plug-ins for Containers, and Red Hat OpenShift Container Platform with the OpenShift Virtualization feature enabled combine to create a powerful and flexible platform for virtual machines and containerized applications.

One of the challenges that many organizations face is to manage separate platforms, one for VMs and another for containers. Now organizations can implement a single platform to run and manage both virtual machines and containers.

With Hitachi Storage Plug-in for Containers and Hitachi VSP One Block storage, your organization can dynamically provision and deliver enterprise shared and persistent storage for virtual machines and containers.

Product descriptions

This section provides information about the hardware and software components used in this solution.

Hitachi Integrated Systems Platform

The Hitachi Integrated Systems platform is a high-performance, low-latency, integrated, converged solution using Hitachi Virtual Storage Platform One Block storage, Hitachi Advanced Server HA820 G3, as well as HA810 G3 with Sapphire Rapids Scalable Processors.

Hitachi Virtual Storage Platform One Block

The Hitachi Virtual Storage Platform One Block series simplifies system setup and management through Hitachi Clear Sight and VSP One Block Administrator. Dynamic Drive Protection reduces RAID complexity, and always-on compression and deduplication enhance simplicity. Virtual Storage Platform One Block with QLC (quad-level cell) delivers the industry's most reliable high density, cost optimized All-Flash Array for read-intensive workloads. It is a compelling infrastructure option for IT organizations that support hybrid cloud environments looking to balance performance, capacity and cost.

Dynamic Carbon Reduction optimizes energy usage by switching CPUs to ECO mode during low activity. Adaptive Data Reduction (ADR) is always on, enhancing efficiency and reducing the overall CO2 footprint.

Thin Image Advanced (TIA) integrates with major snapshot ecosystems, prioritizing security by defending against threats and ensuring data confidentiality. CyberArk Privileged Access Manager plugins enhance block storage system security by prioritizing data confidentiality, ensuring compliance, and actively defending against security threats.

Hitachi Virtual Storage Platform One Block 20 includes 3 dedicated models that support both TLC and QLC NVMe SSD drives, as well as Fibre Channel, iSCSI, and NVMe TCP connectivity. The new capabilities remove complexity: data reduction is always on, Dynamic Drive Protection removes complicated RAID setup, and Dynamic Carbon Reduction delivers real world reduction in power consumption. In addition, the models are FIPS compliant.

- VSP One Block 24 – 256 GB Cache + SW Advanced Data Reduction (ADR) + 24 cores
- VSP One Block 26 – 768GB Cache + 2 × Compression Accelerator Module (CAM) + 24 cores
- VSP One Block 28 – 1TB Cache + 4 × CAM + 64 cores

In short, the Hitachi Virtual Storage Platform One Block series combines simplicity, sustainability, and robust security features to optimize system management, energy efficiency, and data protection.

Hitachi Storage Virtualization Operating System RF

Hitachi Block Storage

Hitachi Storage Virtualization Operating System RF powers the Hitachi Virtual Storage Platform (VSP) family. It integrates storage system software to provide system element management and advanced storage system functions. Used across multiple platforms, Storage Virtualization Operating System includes storage virtualization, thin provisioning, storage service level controls, dynamic provisioning, and performance instrumentation.

Flash performance is optimized with a patented flash-aware I/O stack, which accelerates data access. Adaptive inline data reduction increases storage efficiency while enabling a balance of data efficiency and application performance. Industry-leading storage virtualization allows SVOS RF to use third-party all-flash and hybrid arrays as storage capacity, consolidating resources for a higher ROI and providing a high-speed front end to slower, less-predictable arrays.

See https://www.hitachivantara.com/en-us/products/storage-platforms/storage-software for more information.

Hitachi Advanced Server

Designed to unlock the full benefits of the hybrid cloud, Hitachi Advanced Server models deliver high performance and enhanced security while reducing operational costs. Global enterprises, cloud service providers, and governments trust Hitachi servers to run bare metal, virtualized, or containerized applications. Powered by industry-leading scalable processors, Hitachi servers are ideal to deliver edge, core, and cloud IT services.

Hitachi Advanced Server supports a variety of GPUs for parallel processing which is used in a wide range of applications, including AI, graphics, and video rendering.

Hitachi servers are designed and optimized to maximize performance for VMware, Red Hat, Oracle, bare metal, virtual desktop infrastructure (VDI), SAP, analytics, and other enterprise workloads.

Cisco Nexus switches

The Cisco Nexus switch product line offers a range of solutions that simplify the connection and management of disparate data center resources through software-defined networking (SDN). Leveraging the Cisco Unified Fabric, which unifies storage, data, and networking (Ethernet/IP) services, the Nexus switches create an open, programmable network foundation built to support a virtualized data center environment.

Brocade switches from Broadcom

Brocade and Hitachi Vantara have partnered to deliver storage networking and data center solutions. These solutions reduce complexity and cost, as well as enable virtualization and cloud computing to increase business agility.

Brocade Fibre Channel switches deliver industry-leading performance with seventh generation 64Gb/sec Fibre Channel interfaces, simplifying scale-out network architectures. Get the high-performance, availability, ease of management, and support for the next generation of Hitachi Virtual Storage Platform storage systems on a solid storage network foundation that can grow as your need grows.

See https://www.broadcom.com/products/fibre-channel-networking/switches for more information.

Red Hat OpenShift

Red Hat Enterprise Linux High Availability Add-On allows a service to fail over from 1 node to another with no apparent interruption to cluster clients, evicting faulty nodes during transfer to prevent data corruption. This Add-On can be configured for most applications (both off-the-shelf and custom) and virtual guests, supporting up to 16 nodes. The High Availability Add-On features a cluster manager, lock management, fencing, command-line cluster configuration, and a Conga administration tool.

See https://www.redhat.com/en/store/high-availability-add#%3Fsku=RH00025 for more information.

Migration Toolkit for Virtualization

Migration Toolkit for Virtualization (MTV) enables you to migrate virtual machines from different sources providers to an OpenShift Virtualization destination provider. The following source providers are supported:

- VMware vSphere and Open Virtual Appliances (OVAs) created by VMware vSphere
- Red Hat Virtualization (RHV)

- OpenStack
- Remote OpenShift Virtualization clusters

Red Hat Enterprise Linux

Using the stability and flexibility of Red Hat Enterprise Linux, reallocate your resources towards meeting the next challenges instead of maintaining the status quo. Deliver meaningful business results by providing exceptional reliability on military-grade security. Use Enterprise Linux to tailor your infrastructure as markets shift and technologies evolve.

Hitachi Vantara

Corporate Headquarters
2535 Augustine Drive
Santa Clara, CA 95054 USA

HitachiVantara.com/contact