Red Hat
Consulting

Manufacturing Customer

Virtualization Migration Assessment and High-Level Design

Table of Contents

Preface

Confidentiality, Copyright, and Disclaimer

Revision History

Version	Date	Contributor	Role	Description
0.1	10/01/2024	-	Architect	Initial version
1.0	10/10/2024	-	Delivery Lead Architect	Design and Migration Approach Review
1.1	10/14/2024	-	Delivery Lead Architect	Updates from customer feedback

Participants in the Engagement

Manufacturing Customer Company

Name	Role	Email Address
–	Dir Compute Engineering	–
–	Global Virtualization	–
–	Automation	–
–	Technical Lead	–
–	Storage	–
–	App Deployment	–
–	DB engineering	–
–	Storage	–
–	Network Planning	–
–	Infra Arch	–
–	Edge Compute	–
–	Senior Architect	–
–	Network Planning	–
–	Compute Virtualization	–
–	Virtualization ESX Ops	–
–	Release Management	–
–	EA	–
–	Compute Services	–
–	Security Manager	–

Partner Team Members

Name	Role	Email Address
–	Sr. Architect	–
–	Engagement Lead	–
–	Strategic Account Executive	–
–	Sales Specialists	–
–	Senior Account Solution Architect	–
–	Customer Success Executive	–

Executive Summary

Manufacturing Customer Company has asked Red Hat to provide a direction for migrating its current VMWare-based virtualized workloads to the Red Hat OpenShift Virtualization[1] Platform. These VM workloads are spread across Manufacturing Customer's Data Centers ("DCs") and its facility ("distributed") environments. The Red Hat OpenShift Platform can enable these virtualized and cloud-native workloads by deploying and managing the virtual machines utilizing Kubernetes constructs ("KubeVirt").

Red Hat Services engaged with Manufacturing Customer in October 2024 to embark on this journey by executing the Migration Virtualization Assessment (VMA). This assessment aimed to understand Manufacturing Customer's requirements and environment to determine if OpenShift Virtualization could fit it well. During this assessment, the Manufacturing Customer team indicated its desire to migrate the current virtual workloads (for both DC and distributed) by July 20xx.

This document provides:

- An analysis of Manufacturing Customer's current VM environments and workloads
- Red Hat's recommended migration approach
- High-Level Design (HLD) for the proposed solution to initiate the pilot and Migration Factory
- Requirements for a pilot to validate a jointly selected set of use cases
- Indicative high-level migration schedule and milestones

The scope of this document is limited to the Manufacturing Customer's on-premise deployment of the Red Hat OpenShift Virutualization Engine with the sole purpose of Red Hat OpenShift Virtualization.

Analysis of the current VM footprint (13,735 in-scope VMs) shows a 52::41::7 distribution for the easy::medium::hard migration complexity. This is on par with the workload complexity in similar-sized environments, and the high percentage (7,155, >50%) of "easy" category VMs in the Manufacturing Customer environment provides a high confidence in a streamlined and automated migration.

[1] https://www.redhat.com/en/technologies/cloud-computing/openshift/virtualization

Migration Approach

Given the migration program's complexity and scale, we recommend a two-phase approach consisting of Foundation and Expansion phases.

Foundation Phase

The Manufacturing Customer needs to evaluate the solution proposed and prepare to migrate around 14 thousand VMs as quickly as possible. To accomplish this, we have divided the foundation phase into two stages: a pilot and a production build & migration factory.

Foundation: Pilot Stage

The first stage of this phase is to define and deploy the **Pilot** in a certification environment that will be implemented by the Manufacturing Customer and Red Hat teams. This environment will prove the proposed OpenShift Virtualization architectures' functionality, performance, and reliability as it would be implemented inside Manufacturing Customer's network and data center constraints for both DC and the facility/distributed scenarios.

This certification environment will be based on the architecture proposed later in this document. The pilot will be executed in an internal (on-premise) certification environment at Manufacturing Customer.

In addition, Red Hat proposes to reuse existing OpenShift automation and other reusable artifacts developed by Manufacturing Customer for its existing OpenShift deployments to maximize reuse and enable standardization across the current containerized workloads and the new VM workloads.

Foundation: Production Build & Migration Factory

The second stage of the Foundation phase is the **Production Build & Migration Factory,** which will prepare the final environments (DC, facility/distributed, etc) to receive workloads and create high-level automation to support large-scale VM migration processes. The key objectives of this stage are the following:

- Create **cluster management capabilities and operational procedures** (cluster provisioning, Day-2 Ops) to efficiently manage a fleet of Openshift Virtualization clusters across the different Manufacturing Customer environments.
- Build the **necessary integrations** to enable Manufacturing Customer's existing monitoring, alerting, and marketplace tools to operate with the OpenShift virtualization platform.
- Create and test **automation to migrate** real workloads in a wave/batch pattern, leveraging the Red Hat Migration Toolkit for Virtualization (MTV).
- Create and test the **migration process and governance procedures** to support large-scale migration waves, with concurrent migrations where feasible.
- Create and test the **day-2 governance and operational processes** and practices (including automation) to manage and operate the virtualized workloads post-migration.

- **Train and upskill** Manufacturing Customer's operational teams and relevant stakeholders to manage and operate new virtualized workloads during and after migration.

The foundation for this phase is presented in this document as the High-Level Design (HLD).

Expand Phase

The "expand" phase is when the **large-scale VM migration waves** are planned and executed in a Migration Factory model. The scaled migrations of VM workloads are organized into a series of well-defined migration waves to provide a high-level management structure and oversight of the migration process.

The activities performed during the Foundation Phase (*i.e., Pilot, Production Build & Migration Factory*) are essential to establish the necessary technical capabilities to enable the large-scale migration waves to be planned and executed during the Expand Phase.

This document provides initial estimations around the possible duration of an Expand Phase to implement the migration of the majority of Manufacturing Customer's VM workloads to the Openshift virtualization platform. These initial estimates offer a preliminary plan. A more accurate and detailed implementation plan for the large-scale migration project will only be possible after completing the Foundation phase, which will provide more reliable planning data.

Milestone Estimates

The Red Hat Services team's recommended approach will help the migration team ramp up and execute the migration process quickly, adding knowledge, automation, and planning to the effort.

Assuming a start date for the Foundation Phase of Month x, 20xx, **we estimate the full completion of the migration by June 20xx**. Note that this is an indicative estimate, and this will be refined at the end of the Foundation phase, after incorporation of the schedule constraints, including, but not limited to, Manufacturing Customer's release and change windows.

These estimates assume that any procedural roadblocks can be mitigated with automation and that **no extra effort is made to modernize applications** during the migration phase. Application modernization may still be accomplished as a follow-on effort after the migration program is completed.

Please note that no information was made available regarding Manufacturing Customer's release windows or freeze periods at the time of this assessment. These estimates are based on standard industry practices for quarterly releases and freezes. They should be further refined to align with Manufacturing Customer's specific schedules and must be adjusted for known or planned change windows for both DC and facility/distributed environments[2].

[2] The facilities/distributed environments may have predefined change windows which need to be taken into consideration for schedule refinements.

Assessment Approach

Overview

Relevant teams and executive sponsors at Manufacturing Customer met with Red Hat Consulting for four days of discovery sessions to assess their current VM portfolio. As part of these sessions, the Red Hat team proposed an approach to migrate away from their current virtualization platform and adopt Red Hat OpenShift Virtualization.

Assessment Sessions Delivered

The following sessions were delivered during the four (4) day workshop:

Session Name	Session Outcomes
Assessment Kickoff and Information Gathering Session	Create a shared understanding of target outcomes and key stakeholders, as well as a plan for success for the assessment sessions and the virtualization migration initiative.
OpenShift Virtualization Features	Enhance Manufacturing Customer's understanding of capabilities and features of OpenShift Virtualization and the options available for replacing the current VM platform. Provided a high level Demo of the Virtualization tools.
Virtual Environment Deep Dive	Understand the current environment at Manufacturing Customer and gather requirements for the migration or containerization of VMs to OpenShift Virtualization.
Security and Compliance	Understand security posture, requirements, and their impact on VM migrations and platform configuration.
Storage, Backup and Disaster Recovery	Understand current disaster recovery strategy and identify pain points across applications backup, infrastructure backup, and recovery.
Environment Monitoring and Observability	Understand Manufacturing Customer's Storage requirements. Understand Manufacturing Customer's current environment monitoring strategy and identify pain points across data collection, storage, delivery, visualization, and analytics.
VM Portfolio Analysis	Understand critical factors contributing to migration strategy and identify pilot workloads for migration.

Session Name	Session Outcomes
Migration Factory Approach	Enhance Manufacturing Customer's understanding and propose the migration approach. Gather migration factory requirements
Cloud Strategy	Review Manufacturing Customer's cloud strategy and approach, understand self-service aspects and address cost management
Third Party Integrations (ISV)	Review of Manufacturing Customer's Third Party Integrations
Solution Proposal, Findings and Roadmap Review	Provided High-level architecture proposal, covering various areas of Design drivers and technical decision points. Review of findings: architecture assessment, migration complexity, and timelines. Overview of PoV, production build, and factory migration milestones. Overview of Services Journey for accelerated implementation

Drivers

During the assessment kickoff, Manufacturing Customer stakeholders identified the following drivers for adopting a new virtualization platform:

- **Cost Reduction**:
    - The primary driver is to reduce the overall infrastructure costs associated with the VMware platform. This includes licensing fees, operational costs, and the high cost of maintaining and scaling VMware environments.
- **Resource Optimization**:
    - Manufacturing Customer aims to optimize the utilization of existing hardware resources to ensure maximum efficiency. This includes better management of compute, storage, and network resources to reduce waste and improve performance.
- **Container Strategy**:
    - Manufacturing Customer's CaaS Team has a substantial existing implementation and expertise in the OpenShift platform for container workloads.
    - Manufacturing Customer can leverage this expertise gained over years of operationalizing to quickly adopt virtualization as an additional workload.
    - Support the future modernization of applications from monolithic to microservices, and follow current container architecture and adoption.
- **Operational Efficiency**:
    - Enhancing operational efficiency by automating routine tasks and minimizing manual interventions is another critical objective.
    - This approach involves utilizing existing automation tools, alongside Red Hat Ansible Automation Platform (AAP), to manage deployments and configurations more effectively.

Goals

Along with the previously described drivers, these were the additional goals provided by the Manufacturing Customer team during the Assessment sessions.

- Migrate all feasible VM workloads from VMware by **Month 20xx**.
- Rapid VM migration from current platform while maintaining **high availability**.
    - Migrate with the least amount of impact to DC's and facilities, while considering limitations to bandwidth and hardware at certain global distributed locations.
- **Frictionless Application Workload Migration**:
    - Achieve smooth and minimal disruption during application workload migrations.
    - Where possible, the migration should be transparent from an infrastructure perspective for the application owners (*the application teams will need to validate behavioral correctness, but the underlying platform change should be transparent*)
- **Automation-First Approach** and Consistency of Implementation:

    - Identify capabilities to assist with Manufacturing Customer's goal of establishing consistent Deployment and Day-2 operational automation practices for DCs and facilities.
    - Reuse existing automation and other platform artifacts (e.g., the GitOps based IaC) developed for the current OpenShift CaaS implementation.
- **Optimal Hardware usage**:
    - Reuse existing hardware types for the nodes, where possible.
    - Reuse storage used for existing software defined storage solutions.
    - Prepare for future hardware type of servers and storage.
- **Develop Technical Expertise**:
    - Provide competency skills growth for virtualization engineers, specialists, and support staff.
    - Focus on completing competency skills within 6 to 12 months utilizing the comprehensive Red Hat training options included in the solution.
    - Lean on and learn from Red Hat experiences from previous virtualization efforts.
- **Implement Platform Best Practices**:
    - While the primary priority is on migrating the VMs as-is, use this migration to implement best practices (including automation) where possible ("do it right the first time")

Outputs

Manufacturing Customer stakeholders have requested the following outputs from this assessment:

- A **solution proposal for migration** from the current VMWare implementation to the OpenShift Virtualization platform.
- A Solution proposal for a **pilot deployment** of the OpenShift Virtualization platform within Manufacturing Customer's data centers, to prove and validate a set of use-cases that reflect the current operating environments.
- **Mapping of the current environment** and high-level categorization of VM workloads.
- **Workload analysis** including the relative level of complexity for server and virtual machine migrations based on workload type and other factors.
    - Risk profiles
    - Identify what may not be able to migrate to the target environment (including VM appliances/COTS that might be unsupported by their vendors)
- A **High-Level Design (HLD)** of the target OpenShift Virtualization environments with recommendations for addressing potential pitfalls.
- A **high-level roadmap/estimated migration timeline** with a path to completion before Manufacturing Customer's Month 20xx deadline.

Current Environment

Infrastructure: DC, Manufacturing facilities & Other Sites

Manufacturing Customer runs x data centers around the world:

- x in North America (NA),
- x in Europe, and
- x in Asia.

These Data Centers (DC) are managed by the IT teams in charge of the infrastructure: network, storage, and compute.

Manufacturing Customer also has extensive virtualization operations in multiple manufacturing facilities ("distributed" sites) across the world, hosting workloads that are critical to its manufacturing process.

A few other virtualization environments are deployed in other sites, such as:

- Product development sites
- Warehouses (non-manufacturing)
- Call centers

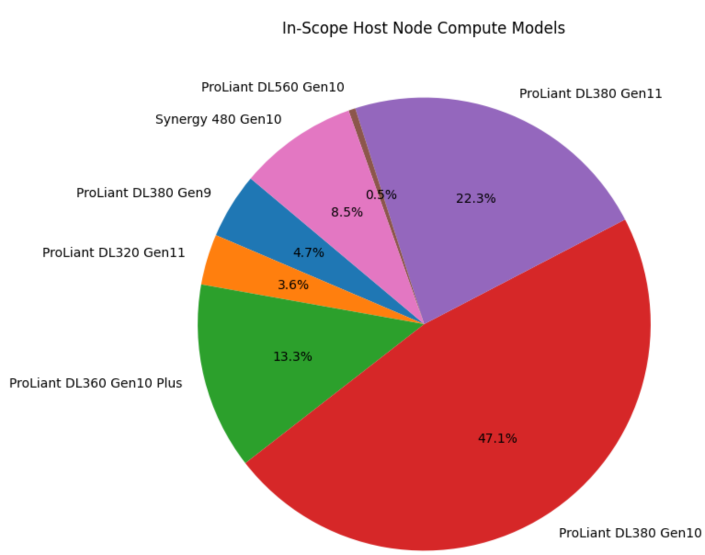
The current Data Centers in NA are divided into three (3) halls each providing three (3) failure domains. However Hall1 is being phased out, which gives a total of four (4) failure domains between these two (2) Data Centers which are connected with fiber. Both light and dark links exist.

For Storage each data center is considered a region for storage and a hall is a failure domain.

Virtualization Hardware

The virtualized hosting environment consists of a mix of HP and HPE hardware which is certified for all current versions of Red Hat Enterprise Linux (RHEL) and current version of Red Hat OpenShift Virutualization Engine:

- HP ProLiant DL380 Gen9
- HPE ProLiant DL320 Gen11
- HPE ProLiant DL360 Gen10 Plus
- HPE ProLiant DL380, Gen10, Gen11
- HPE ProLiant DL560 Gen10



- HPE Synergy 480 Gen10

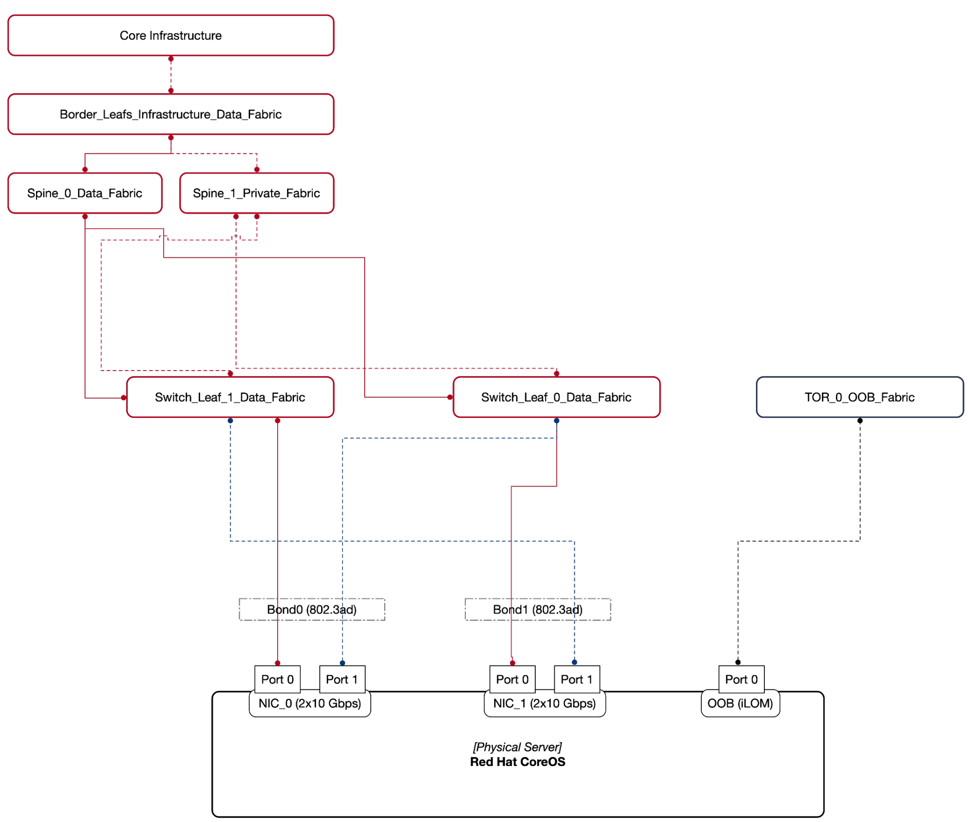
All ESXi hosts boot from local disk or NVME: this is dependent on when the hardware was purchased and what options were available.

The DC storage is provided by Fusion SDS for current OpenShift clusters for CaaS, and NetApp providing NFS v3 to Current ESXi hosts.

Virtualization network

Networks inside the Data Centers are in a spine-leaf configuration using Cisco ACI fabric. The factories are spine leaves that have end of row switches connected back to the factory core. New factories are top of rack switches back to factory core. All networking hardware is from Cisco.

EPGs and VLANs are used to isolate traffic and storage run over the same network hardware as the rest of the environment.



Virtualization storage

Storage in the data centers is set up with failure domains to provide maximum resilience. NetApp provides the backing storage for ESXi LUNs over NFS v3and provides backups leveraging volume snapshotting.

- Local storage is leveraged for ESXi hosts to boot. Currently a mix of flash drives and NVME is used. The differences arise from the purchase date, as the hardware used different storage media. There is no boot from SAN or SD card
- Clusters that are not using NetApp are using “localstorage”
- ODF is used in the current OpenShift (CaaS) environments
- Cohesity was discussed for incremental backups (CBT). RedHat is actively discussing with Cohesity (a partner of Red Hat). and while there is currently no ETA, this is being actively pursued for an accelerated implementation and general availability[3].

Day-1 and Day-2 operations

This section describes the current technologies and practices used for the existing VMWare-based VM operations.

User management

Users have a principal stored and groups in Microsoft Active Directory. Hashicorp Vault is used for secrets management for CaaS (Containers on OpenShift). The Least-Privilege model is used along with RBAC for resource access and permissions.

Log management

Manufacturing Customer uses Splunk and Google object buckets, where applications send their logs.

Monitoring

Manufacturing Customer uses VMWare vRealize Ops Manager, Shell Monitoring, Dynatrace, QRadar, and Progress Whats Up Gold[4] for monitoring VMs and applications.

Storage uses NetApp Insights for monitoring.

---

[3] Red Hat strongly recommends that Manufacturing Customer also use their relationship with Cohesity to influence the acceleration of this implementation.

[4] https://www.whatsupgold.com/

Alerting for current OpenShift Clusters is performed by Alertmanager to BMC and webhooks to Webex Teams.

Automation and Self-Service

Manufacturing Customer has a semi-automated self-service model for provisioning VMs in the current platform.

The end users fill in the request details in an Excel base spreadsheet template and submit this to the VM operations team for provisioning.

Manufacturing Customer utilizes an **in-house developed tool** that scrapes this request spreadsheet and extracts the relevant information. It also performs data validation and other checks, and converts the request into a machine-consumable JSON.

The tool uses the scraped data and makes API calls to the various systems (including VMWare automation) to create the pre-requisites and execute the blueprints to build the VM. This is common to both DC and facility/distributed space, except that for facilities, the site IT team is also involved in the final build and provisioning.

Manufacturing Customer has a documented standard ("Principles of Virtualization"), which includes the limits, best-practices, and other relevant considerations. Any VM request that does not comply with these principles is reviewed and approved via a separate manual process.

Manufacturing Customer is exploring enhancements to this process and tooling.

Workload Migration: Complexity Analysis

As a part of the assessment, Manufacturing Customer provided a **detailed inventory of the current VM and underlying VMWare infrastructure**[5]. Red Hat has analyzed this data to assess the scope and volume of VM migration candidates, and the complexity of migration.

The information that was shared allowed Red Hat to get a better understanding of Manufacturing Customer's VMWare environment and the complexities in the current virtualization solution. We have identified areas where Red Hat can assist and facilitate Manufacturing Customer to migrate from VMWare to OpenShift Virtualization.

Items that will take more time and care in planning are the **facilities / distributed sites** as these sites are high impact if the site is down, and some sites are constrained on bandwidth as well. Even though it may be technically easy to migrate a VM at these sites, the constraint is the scheduling and business decisions around scheduling the migration windows.

Other difficult migrations will be around the **MS-SQL servers** due to the nature of "Always On" and there are also licensing concerns (for vendor/COTS vendors) that may present some issues during migrations. We have also identified the witness site as a challenging migration, not due to technical complexity, but getting the time window to migrate the VMs

Oracle VMs also pose their own issues where Oracle itself does not support their product running on any virtual platform except for their own (which is a derivative of Red Hat's platform), and VMWare. Red Hat does not see any technical issues but there are licensing and support issues that will need to be addressed.

There are some discrepancies in the data that was provided from the RV tool export, which may skew the analysis. This is caused when a VM is created and not set to the proper OS type and or version in the VM metadata. When vmware tools are installed in the guest, it will report the correct OS version, however this is not always possible: for example with appliances.

Existing Virtual Machines

The table below lists the current VMs across all VMWare clusters and all regions. Note that this includes a set of excluded VMs for vCenter instances that are being decommissioned. Subsequent sections take this exclusion into account.

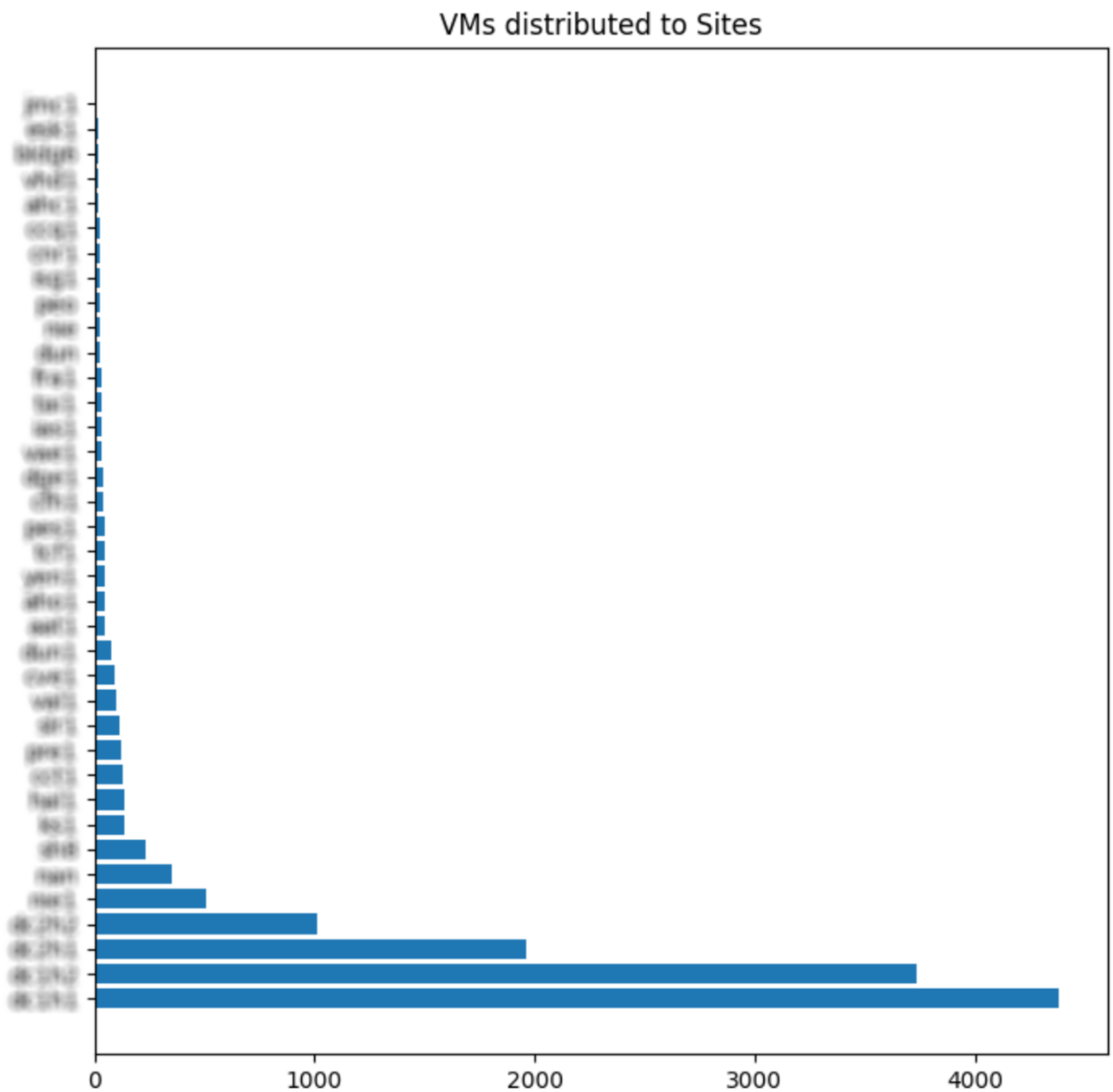
Location	Region	VMs #	vCPU #	vMem (GB)	vDisk (TB)
DCs NSX -	-	320	2,158	24,596	209.4

[5] Provided as RVTools exported spreadsheet reports

Location	Region	VMs #	vCPU #	vMem (GB)	vDisk (TB)
-	-	4,223	243,742	158,850	3,609
-	-	3,802	21,136	159,454	3,209
-	-	1,959	12,010	104,360	1,743
-	-	1010	6896	421,123	991
- Witness Site	-	16	70	344	2.92
-	-	1	4	8	0.141
-	-	91	287	947.6	21.1
-	-	30	83	270	7.819
-	-	32	85	315.9	7.911
-	-	103	462	1,784	36.14
-	-	3	3	2.3	0.095
-	-	60	273	1,049	27
-	-	200	788	2,565	59.46
-	-	70	185	834	17.96
-	-	41	119	399	11.63
-	-	107	465	1,767	37.96
-	-	101	408	1,242	28.47
-	-	139	590	2,187	50.38
-	-	81	277	943	61.87
-	-	322	1094	4,361	108.47
-	-	39	108	344	9.89
-	-	23	57	492	19.47
-	-	44	115	410	9.76
-	-	70	242	762	24.85

Location	Region	VMs #	vCPU #	vMem (GB)	vDisk (TB)
-	-	53	236	927	17.91
-	-	30	79	266	6.93
-	-	24	62	201	7.02
-	-	123	449	1,138	28.03
-	-	46	135	557	13.35
-	-	33	87	303	8.25
-	-	47	146	549	14.18
-	-	39	110	336	11.21
-	-	36	348	5,244	21.98
-	-	73	258	1,550	31.55
-	-	142	551	1,488	36.24
-	-	30	141	1,429	33.14
-	-	264	1248	8,902	22.31
-	-	112	371	1,201	31.94
-	-	49	268	3,528	1734
-	-	25	197	2,994	16.32
-	-	20	50	205	3.78
-	-	32	103	388	8.63
-	-	50	145	586	19.17
-	-	47	131	516	13.66
-	-	86	281	916	27.01
-	-	94	294	1,198	35.93
-	-	14	27	123	2.7

Location	Region	VMs #	vCPU #	vMem (GB)	vDisk (TB)
-	-	47	112	507	14.19
-	-	34	104	334	8.17
-	-	95	279	1,370	28.78
-	-	44	144	442	14.46
-	-	123	437	1,516	54.38
-	-	20	69	111	3.83
-	-	33	83	214	11.23
-	-	22	73	123	4.48
-	-	21	66	285	16.47
-	-	128	455	1,725	55.69
-	-	38	97	393	10.03
-	-	21	148	2,289	1.004
-	-	7	20	78.08	1.06
-	-	133	671	4,036	15.23
-	-	231	1131	6,270	23.74
TOTALS		15,441	301,263	943,648	12,712.68



Distribution of VMs by vCenter

The table below describes the distribution of VMWare VMs categorized by the vCenter instances. This also depicts the following characteristics of the current deployment:

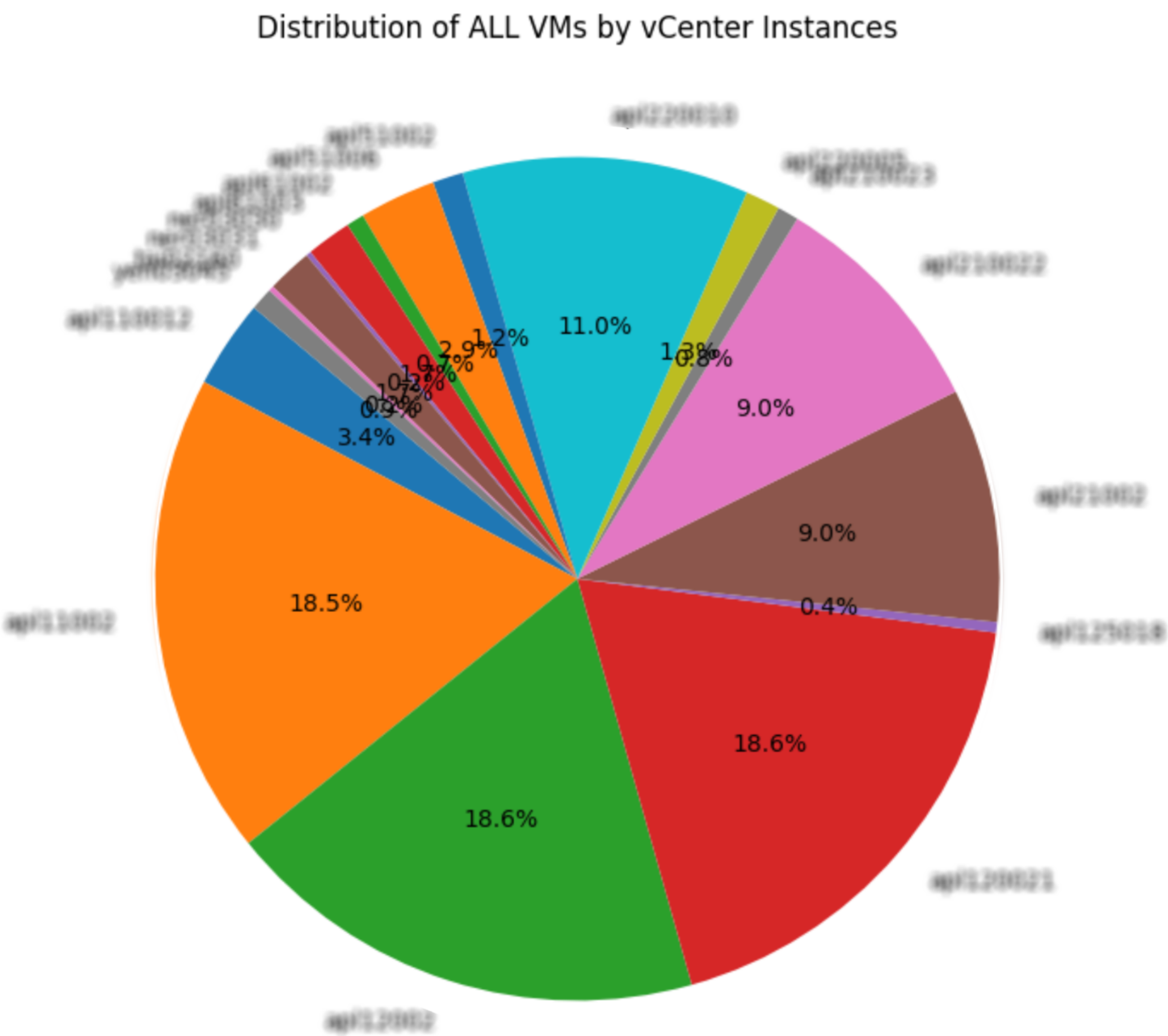
- Total count of host CPUs (**1,613**)
- Total cores in use (**35,736**)
- Host distribution by vCenter Instance (**total 848 hosts**)

vCenter	vCPUs	Memory (GB)	NICs	Total disk capacity (TB)	VM	# CPU	# Cores	# VMs total	Host
-	3,045	16,359.93	707	425.51	677	120	2,136	677	60
-	21,846	155,132.29	3,803	3,395.50	3722	220	6,928	3722	110
-	21,136	159,454.22	3,802	3,209.17	3738	340	8,864	3738	170
-	345	810.81	107	5.05	81	8	104	81	4
-	10,881	91.857.72	1,843	1,742.03	1794	172	4,664	1,794	86
-	1,129	12,502.60	13	1.33	165	22	656	165	11
-	1,649	4,734.97	693	56.46	268	106	1,600	268	53
-	6,896	42,129.34	1,076	990.80	1010	150	3,360	1,010	75
-	1,131	6,270.50	272	237.48	231	39	920	231	28
-	1,798	13,031.60	470	179.31	481	123	1,896	481	72
-	671	4,036.60	157	179.31	133	36	848	133	25
-	202	1,773.56	28	35.92	28	16	184	28	8
-	1,421	10,043.39	328	250.55	328	61	920	328	33
-	3,590	24,245.64	848	305.93	892	173	2,288	892	93
-	103	388.48	29	8.62	32	8	32	32	4
-	433	1,829.37	149	52.83	155	19	336	155	16
Total	**76,276**	**544,601.08**	**14,325**	**11,048.92**	**13,735**	**1,613**	**35,736**	**13,735**	**848**

Approach

Categorizing workload complexity will help with prioritizing the migration plans and provide an estimate of the effort. We base our evaluation on the following:

- Workload environment
- Operating System and Version



- Resource Capacity and Requirement
- Disks per VM and disk size

Workload Environment

We analyzed **13,735 virtual machines** from Prod, Non-Prod, and Development. This is based on the VM inventory data provided by Manufacturing Customer as RVTools exported reports.

The analysis does not include CDC environment as this was not provided by Manufacturing Customer for this assessment.

This analysis does not include the following vCenters due to decommissioning in progress:

- REDACTED

Out of these 13,735 eligible VM workloads from VMware, we first categorized the VMs by the support level of the guest operating system (See https://access.redhat.com/articles/973163 and https://access.redhat.com/articles/4234591), followed by the disk size, as these are the critical factors for migration complexity.

The distribution of DC to facility/distributed site VMs is as follows:

- Total In-scope VMs in DCS: **11,090 (80.74%)**
- Total In-scope VMs in facilities: **2,645 (19.26%)**

Linux Workloads

RHEL and RHEL-derivatives such as CentOS, Rocky and Oracle Linux that are newer than RHEL 5 are categorized as easy migration candidates. These RHEL versions are heavily tested and officially supported by Red Hat (RHEL) or are binary-compatible derivatives (CentOS, Oracle Linux, etc). The older versions will require upgrades before migration and hence are classified as medium complexity.

SUSE is also a supported distribution starting with SLES version 12+, including third-party vendor support.

Debian and Ubuntu are not supported distributions at the time of this assessment, but are in Technology Preview for the platform, hence these are classified as hard complexity. In the case of Ubuntu, Canonical offers commercial support to a few of their operating systems in our platform as stated in the previously mentioned KB article (#4234591).

Various Linux VMs with unlabelled distribution will need to be determined by other means. The effort level will be determined on a case-by-case basis. Many of these will be appliances that require coordination with the appliance vendor for a fully-supported solution.

Operating Systems	Easy	Medium	Hard
CentOS 4, 5, 6,7 (not		441	
CentOS 7	247		
Debian 5,6,7 (not supported)			50
Debian 8, 9, 10,11 (not			90
FreeBSD (64-bit) (not			1
Oracle 7 (not supported)			2
Other (32-bit)(not supported)			32
Other (64-bit)(not supported)			23
Other => 2.6.2 Linux (32-bit) (not supported)			11

Operating Systems	Easy	Medium	Hard
Other => 2.6.2 Linux (64-bit) (not supported)			448
Other Linux (64-bit) (not supported)			9
RHEL 5 (64-bit) (not		4	
RHEL 6, 7, 8, 9 (64-bit)	535		
SUSE 8/9 (32-bit) (not supported)			2
SUSE 10, 11 (64-bit) (not supported)			177
SUSE 12, 15 (64-bit)		4,085	
Ubuntu (64-bit) (not			70
Total	**782**	**4,530**	**915**

*Note: The Oracle Witness VM will be of hard complexity

Windows Workloads

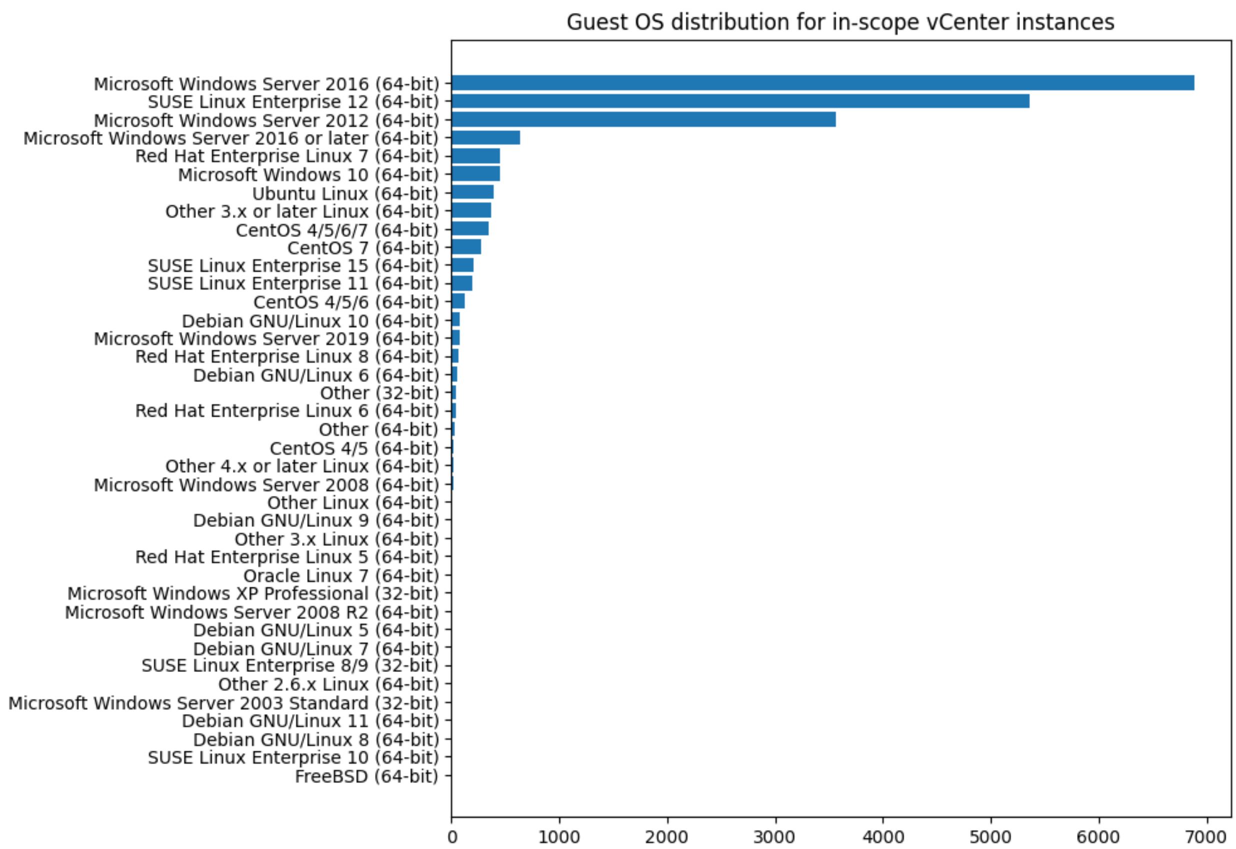
Microsoft Windows Server operating systems are certified by Microsoft through the SVVP[6] program to run on Openshift Virtualization's platform (running on the KVM hypervisor).

Red Hat extends support to Windows client operating systems based on the corresponding server OS version (i.e. Windows 10 and Server 2019). During migration, the Migration Toolkit for Virtualization will automatically install virtio drivers to ensure optimal performance.

Operating Systems	Easy	Medium	Hard
Windows 10 (Manufacturing/ DXD)			585
Windows Server 2016, 2019 *1		5,665	223
Windows Server 2008, 2012 *1 *2 (not supported)			3,057
Windows Server 2003 (32-bit) (not supported)			2
Windows XP (not supported)			4
Total	**-**	**5,888**	**3,648**

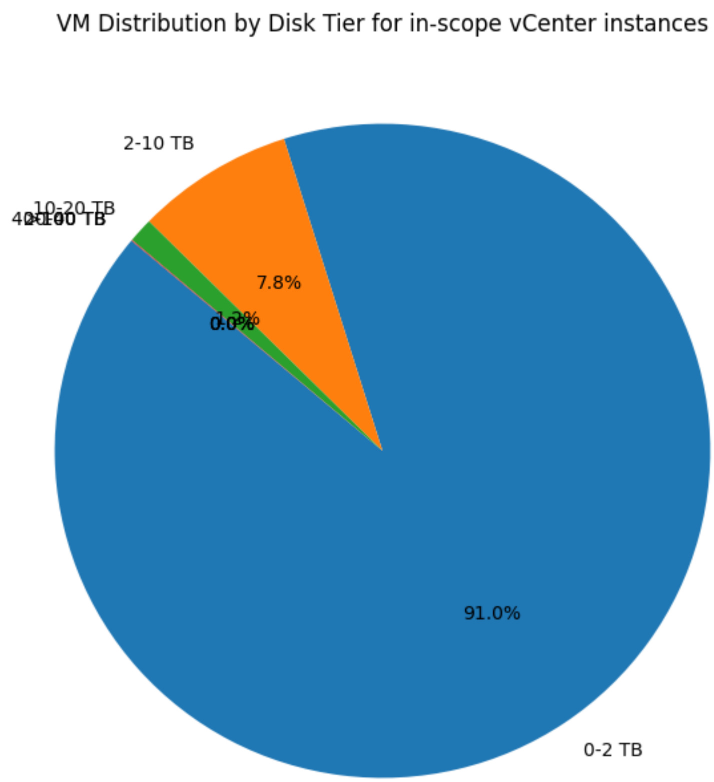
1. SQL ALWAYS ON, will be hard (vm tools reported Windows 2012 as 2016 so was counted as 2016)
2. Windows 2012 R2 supported =<4.13

[6]SVVP: Microsoft Server Virtualization Validation Program (https://www.windowsservercatalog.com/svvp/program-home)


Guest OS distribution for in-scope vCenter instances
Microsoft Windows Server 2016 (64-bit)
SUSE Linux Enterprise 12 (64-bit)
Microsoft Windows Server 2012 (64-bit)
Microsoft Windows Server 2016 or later (64-bit)
Red Hat Enterprise Linux 7 (64-bit)
Microsoft Windows 10 (64-bit)
Ubuntu Linux (64-bit)
Other 3.x or later Linux (64-bit)
CentOS 4/5/6/7 (64-bit)
CentOS 7 (64-bit)
SUSE Linux Enterprise 15 (64-bit)
SUSE Linux Enterprise 11 (64-bit)
CentOS 4/5/6 (64-bit)
Debian GNU/Linux 10 (64-bit)
Microsoft Windows Server 2019 (64-bit)
Red Hat Enterprise Linux 8 (64-bit)
Debian GNU/Linux 6 (64-bit)
Other (32-bit)
Red Hat Enterprise Linux 6 (64-bit)
Other (64-bit)
CentOS 4/5 (64-bit)
Other 4.x or later Linux (64-bit)
Microsoft Windows Server 2008 (64-bit)
Other Linux (64-bit)
Debian GNU/Linux 9 (64-bit)
Other 3.x Linux (64-bit)
Red Hat Enterprise Linux 5 (64-bit)
Oracle Linux 7 (64-bit)
Microsoft Windows XP Professional (32-bit)
Microsoft Windows Server 2008 R2 (64-bit)
Debian GNU/Linux 5 (64-bit)
Debian GNU/Linux 7 (64-bit)
SUSE Linux Enterprise 8/9 (32-bit)
Other 2.6.x Linux (64-bit)
Microsoft Windows Server 2003 Standard (32-bit)
Debian GNU/Linux 11 (64-bit)
Debian GNU/Linux 8 (64-bit)
SUSE Linux Enterprise 10 (64-bit)
FreeBSD (64-bit)
0
1000
2000
3000
4000
5000
6000
7000

Disk Size

The size of the VM disk affects the time to copy the vdmk to the cluster and the time it takes to convert the disk. These are controlled by the migration network bandwidth and the CPUs available during conversion.



Disk Size (TB)	Easy	Medium	Hard
1-2 TB	14,270		
3-9 TB		1201	
10-20 TB			163
21-40 TB			6
Total	14,270	1201	169

Virtualization and Modernization Analysis

Out of the total analyzed workloads, there are three main categories:

- Workloads that will migrate to a new virtualization platform (aka OpenShift Virtualization)
- Workloads that currently run in a container management platform virtualized on vSphere and could move to a modern container platform (OKE / OpenShift)
- Workloads that run in an application platform virtualized on vSphere and could move to a modern application platform (OpenShift Platform Plus)

Here is the count of virtual machines associated with these three scenarios:

Use Case Modernization	Count (# of VMs)	# of dedicated hosts
Virtual Machines (general use)	11,487	600
Container Management use case	1480	148
Application Platform use case	1000	100

VDI

Manufacturing Customer is currently running VDI (Virtual Desktop Infrastructure) workloads but has indicated that these are out of scope for this migration assessment.

Virtual Appliances

There are virtual appliances used in use at Manufacturing Customer. However, appliance-specific information (e..g, type of appliance and the application) has not been provided during the assessment. Appliances will require a case-by-case approach and complexity will be based on workload.

The following table is a template for capturing the appliance characteristics for further analysis.

<table>
<tr><th>Purpose</th><th>Name</th><th>Version</th><th>KVM Appliance</th><th>Notes</th></tr>
<tr><td rowspan="2">Disaster Recovery</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>

Backup				
CISO				
SDN				
Storage				
Infrastructure				

Summary of the Complexity Analysis

Categorizing workload complexity helps prioritize the migration plans and gives an estimate of the effort. The analysis shows a **52::41::7 of easy::medium::hard distribution**[7]. The estimation is based on the following:

- Workload environment (prod, non-prod, lab)
  - Lab and non-prod are migrated first to lay the groundwork for production
- Operating System and Version
  - Newer than RHEL 5 = easy
  - RHEL 5 = medium
  - Debian and Ubuntu = hard
  - Windows = medium
- Workload Type (Database, AppServer, Message Bus)
  - Database/Message Bus/Hadoop = medium
  - AppServer = easy
- Resource requirement: CPU, Memory
  - Consideration during migration to not overload clusters
- Size of Disk, Utilized Disk
  - 90% of VMs use < 2TB = easy

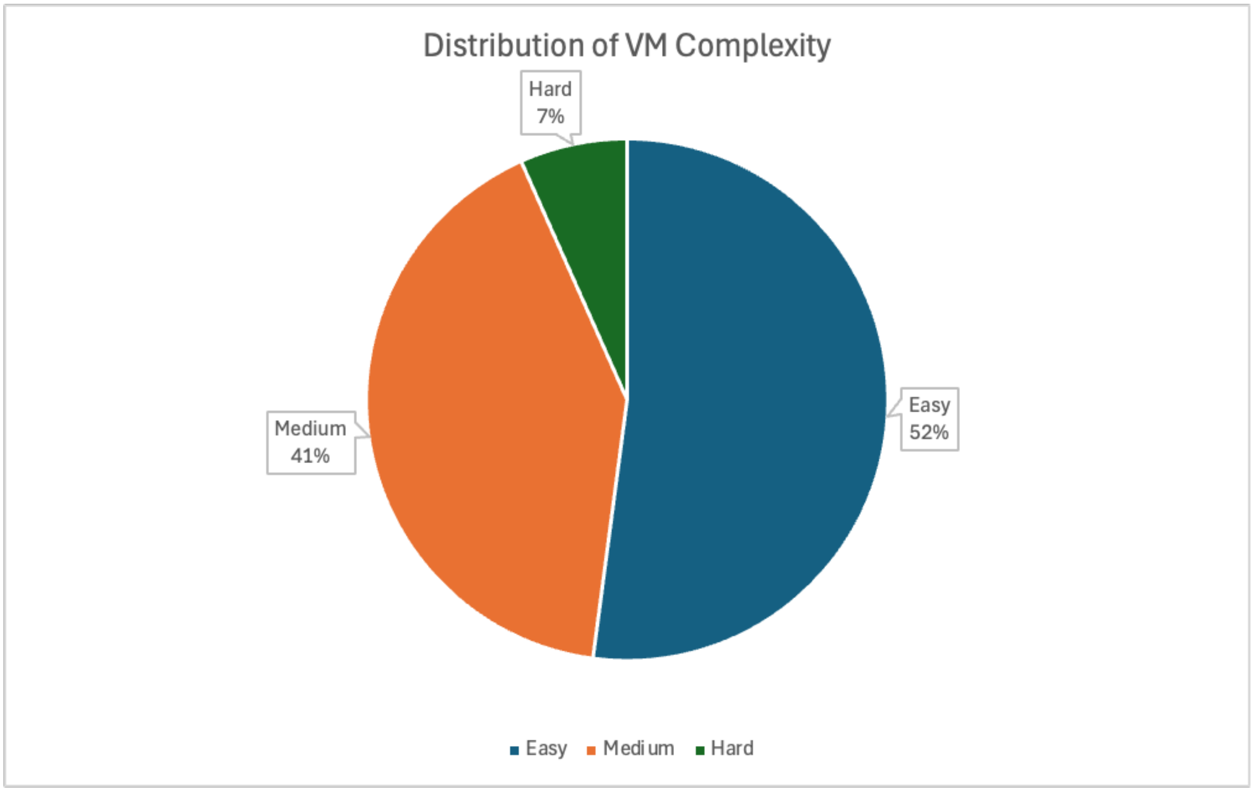
Based on the multiple criteria and parameters listed above, the indicative workload complexity for the 13,735 VM migration candidates is as follows:

Complexity	VM Count	Percentage of Total
Easy	7,155	52%
Medium	5,665	41%
Hard	915	7%
Total	13,735	100%

Note: This analysis is based on the VM inventory and high-level workload analysis provided during the assessment. For the “medium,” and especially the “hard” complexity VMs, additional analysis and characterization will be performed during the migration.

This is on par with the workload complexity in similar-sized environments, and the high percentage **(5,155, >50%) of “easy” category VMs** will make a significant portion of the migration amenable to automation.

[7] As percentage of the in-scope VMs for migration; based on VM inventory data from Manufacturing Customer



- The size of the drives impacts complexity as it will increase the amount of time required to migrate the VM from one platform to another. This will also keep more traffic on the network than smaller drives,
- Note: due to unavailability of the appliances, unidentified Linux, and mislabeled Windows workload information, the complexity driven by these VMs has not been applied to the analysis above.
- The difficulty categories (easy, medium, and hard) are not just tied to a technical profile. This can also be due to the ability to get a maintenance window and its impact on services the VM is providing as it is moved to the new platform.

Proof of Value Pilot: Use Cases

As part of the assessment, the need to validate a set of use cases for both the DC and the facility/distributed scenarios, was identified. Red Hat and Manufacturing Customer stakeholders jointly identified a number of key scenarios and use cases that will be validated in a pilot OpenShift Virtualization environment.

The use cases (categorized by the environment/type) are as follows:

Environment	Use Case	Description
For both DC and facility/distributed	Incremental Backup and Restore Using Cohesity (CBT)	○ Work with Cohesity to address desire for incremental backup capability and test. ○ Prove out the restore function from backup
For both DC and facility/distributed	Monitoring	○ Review monitoring and observability capabilities to understand the expected performance impacts or capabilities during VM migrations.
DC	Implement the "Stretch" cluster for NSX workloads	○ As proposed in the stretch cluster design, to promote and validate the stretch cluster solution for NSX workloads.
DC	Migrate Oracle DB (on SLES)	○ Prove the technical feasibility of running Oracle DB on OpenShift Virtualization for non-production workloads. ○ Work towards resolving the licensing issue with Oracle DB on non-VMWare hypervisors
DC	Validate resolution of current issues with the Migration Toolkit for Virtualization (MTV)	○ Validate MTV is working as expected by resolving the list of open issues Manufacturing Customer has for MTV.

Environment	Use Case	Description
DC	Assess auto-scaling of VMs in the new platform	○ Determine the feasibility, constraints, and approach for auto-scaling VMs on OpenShift Virtualization ○ Identify the specific set of Manufacturing Customer VMs that currently use VM auto-scaling on VMWare and need to retain this capability on the new platform
facility/distributed	Source IP Preservation	○ Validate the source IP preservation during migration of VMs in the facility/distributed environment.
facility/distributed	Assess uplink bandwidth constraints	○ Validate and demonstrate the migration capabilities against Manufacturing Customer's known bandwidth constraints, mostly with facility/distributed locations.

The proposed program approach provides details of this pilot phase and a subsequent section provides details on the OpenShift Virtualization pilot environments that need to be set up in Manufacturing Customer's on-premise data centers.

Red Hat OpenShift Virtualization: High-Level Design

OpenShift Virtualization uses the Linux kernel hypervisor, **Kernel-based Virtual Machine (KVM)**[8], to bring virtual machines to OpenShift as native objects. OpenShift's features and capabilities support cluster management, scalability, and virtual machine consumption.

KVM is a core component of the Red Hat Enterprise Linux kernel and has been used in production for more than 15 years in other hypervisor products, such as Red Hat Virtualization, Red Hat OpenStack Platform, and Red Hat Enterprise Linux. OpenShift uses the same KVM, QEMU, and libvirt as those platforms; it is a robust, stable, and scalable type-1 hypervisor for virtualization workloads.

Four (4) deployment architectures have been designed on the basis of the findings from the assessment. The table below describes the specific scenario each architecture addresses, along with its characteristics. All these designs are deployed on bare-metal servers.

Architecture	Intent / Target Environment	Key Characteristics
DC	For the majority of the DC based VMs. Storage is not shared across the two DCs	- Master/control and infra planes are distributed across both data centers and across three availability zones (2+1) for quorum and resiliency - The cluster topology in DC1 is mirrored in DC2 - VMs in each data center operate independently of the other DC - Storage is not shared between the two data centers, and can use CSI drivers - For a DR scenario, failover between data centers can be automated and could utilize a Global Load Balancer (GLB) - This architecture is similar to the existing OpenShift (for containers) deployment topology at Manufacturing Customer (CaaS)

[8] https://www.redhat.com/en/topics/virtualization/what-is-KVM

Architecture	Intent / Target Environment	Key Characteristics
“Stretched” DC for NSX workloads	For the existing VMs utilizing the VMWare NSX feature. These VMs run in the DCs but serve facility/distributed workloads.	- Master/control plane is distributed across both data centers and three availability zones (2+1) for quorum and resiliency - Infrastructure nodes are distributed across both data centers and four availability zones (2+2) - Storage is provided by IBM Fusion SDS, that is “stretched” across both data centers and all availability zones - This architecture replaces the existing VMWare NSX capabilities for resiliency and High Availability (HA)
facility/Distributed	For VMs that run inside the facilities (distributed environments)	- This architecture runs within a single, logical DC inside the facility - Resiliency is established by distributing the control and infrastructure nodes across separate racks - Network bandwidth constraints on the WAN are taken into account
Standalone	For VMs that run in a single-node environment. Many of these VMs provide network services.	- This architecture uses Single Node Openshift (SNO) topology and runs on a single bare-metal node.

We recommend that when deploying OpenShift Virtualization, failure domains are defined at the physical architecture layer when possible to provide better resiliency. Topology labels can be applied to nodes and used by VMs and containerized applications to spread across failure domains. A few examples of spreading across failure domains are:

- Use a multi-rack topology (at least 3), distributing the roles across the multiple racks
- Each rack is a failure domain
- Deploy a CLOS Leaf-Spine architecture, where each rack is contained within a Leaf

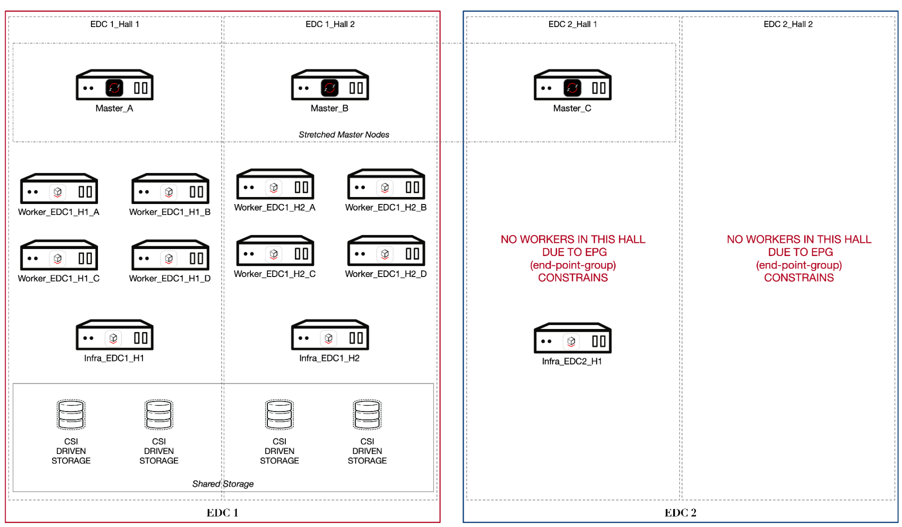
For more information, check the following references:

1. https://kubernetes.io/docs/concepts/scheduling-eviction/topology-spread-constraints/#node-labels
2. https://kubernetes.io/docs/reference/labels-annotations-taints/#topologykubernetesioregion

DC Architecture

The data center (DC) OpenShift Virtualization architecture leverages the current OpenShift design used for container workloads at Manufacturing Customer (CaaS):

Virtualization High Level Design
Default Virtualization Environments

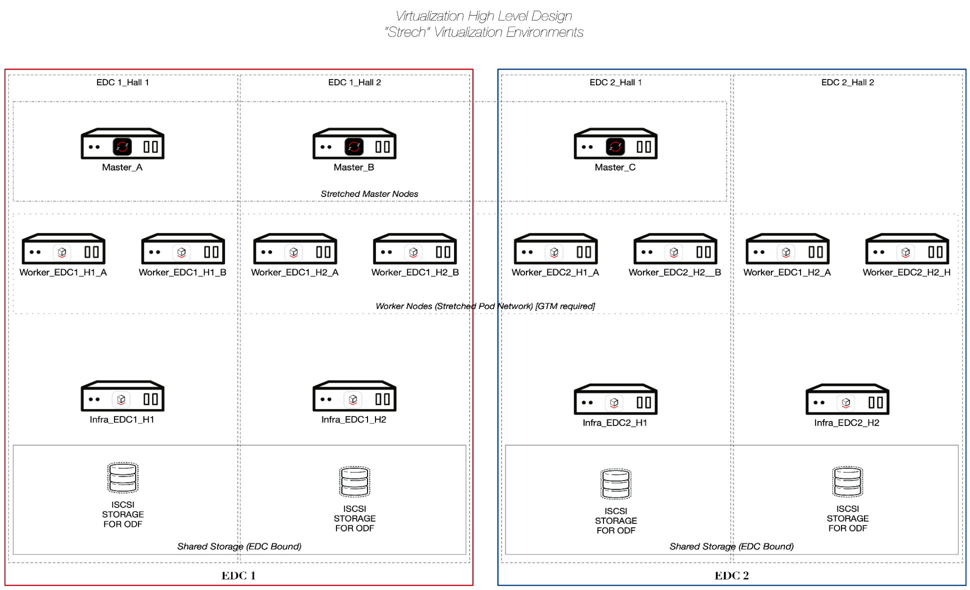


- The DC1 OpenShift Virtualization cluster is “stretched” across the two data centers (the master and infra nodes are stretched)
- This topology is mirrored in the second datacenter, DC2 (not depicted in the diagram above)

- Note: Manufacturing Customer has enquired whether a 4th master node is desirable. This is not available in the current OpenShift product versions but will be a in a future release

“Stretched” DC Architecture

The following diagram illustrates a “stretched” OpenShift Virtualization cluster where storage (using Fusion SDS) is stretched across both data centers. This design is specifically for VM workloads that currently use the VMWare NSX feature.

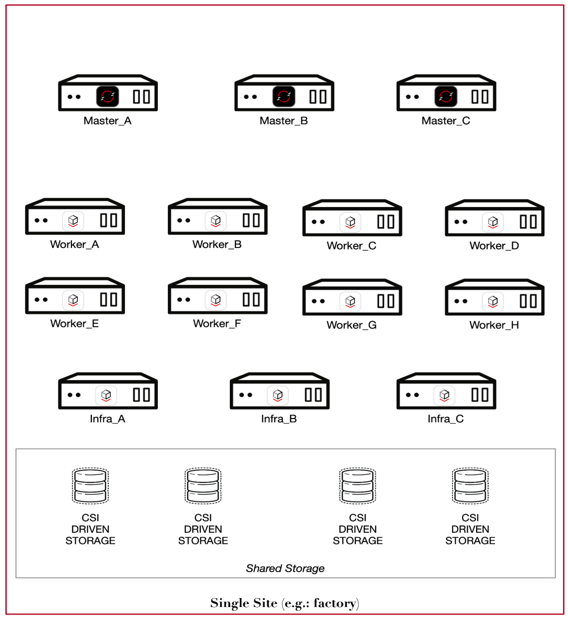


- Due to the very low latency from DC1 to/from DC2 this allows for a stretch cluster
- This allows two (2) nodes across the two (2) failure domains giving higher resilience
- Data is replicated across the stretch cluster to both data centers using Fusion SDS

Facility/Distributed Architecture

Design for OpenShift Virtualization.

Virtualization High Level Design
Single Site Virtualization Environments

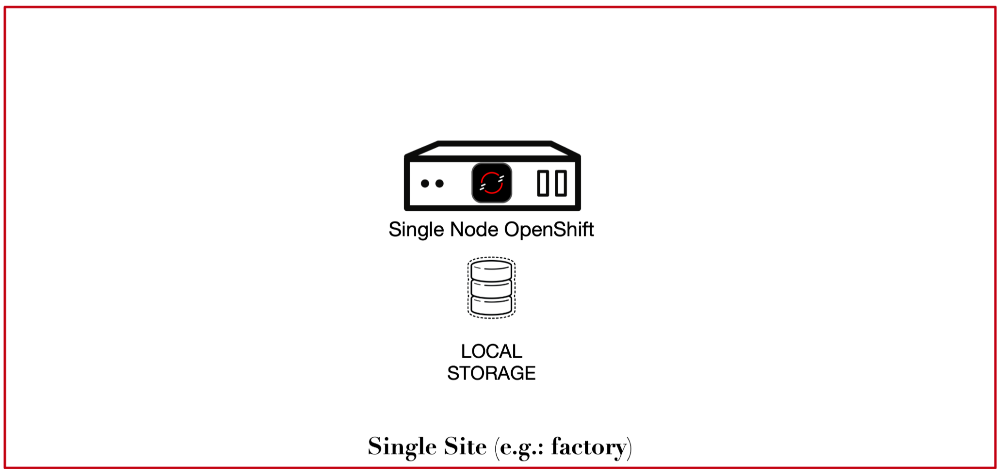


- Facility / Distributed architecture providing OpenShift platform single site availability but leveraging distribution between racks to provide some "Rack redundancy"
- Facilities' uplink connectivity / bandwidth availability will need to be taken into consideration per site, as each site has different connectivity
- Source IP needs to be preserved for a number of facility/distributed VM workloads
- Recommend 1 Gbps WAN for installation of the cluster. If the connectivity is slower, then leveraging disconnected install and maintenance is recommended

Standalone Architecture

This architecture replaces the single ESXi hypervisors that are standalone and provide network services. In this scenario, these ESXi clusters will be replaced with Single-Node OpenShift (SNO)[9] clusters.

Virtualization High Level Design
Single Node OpenShift (SNO)



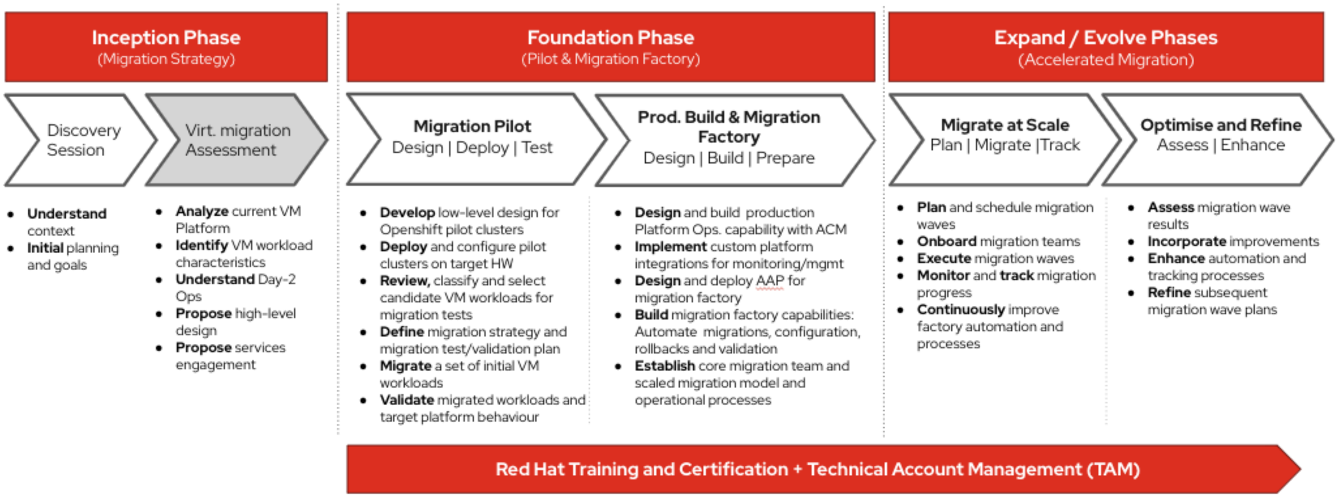
- In this architecture, we recommend to use local volumes by enabling the Local Storage Operator.
- When using a Single Node Openshift considerations will need to be taken around resource planning (cpu/memory requirements to run the kubernetes services).

[9] https://www.redhat.com/en/blog/meet-single-node-openshift-our-smallest-openshift-footprint-edge-architectures

Recommended Program Approach

Introduction

Red Hat recommends the following, proven, migration methodology to migrate a large number of VMs to the Openshift Virtualization platform. The diagram below illustrates the phased project model developed by Red Hat to support large scale migration projects to OpenShift Virtualization.



Red Hat Openshift Virtualization Migration - Project Model

The Red Hat recommended migration approach consists of three major project phases:

- Inception Phase
- Foundation Phase
- Expand Phase

In addition, due to differences in the change/release windows and the deployment architecture for the DC and facility/distributed sites at Manufacturing Customer, **Red Hat recommends two separate workstreams for DC and facilities/distributed**, each addressing these two target site categories, while retaining the overall structure and phases of this migration factory methodology.

Inception Phase

This **Virtualization Migration Assessment report** represents the technical findings of the inception phase, which occurred between **Red Hat Services and Manufacturing Customer** technology teams.

The Inception phase initiates the Red Hat Services customer engagement, and its main objective is to gain an understanding of the customer's business objectives/drivers/constraints, to review and assess the current technical environment and virtualization landscape/workloads, and define a strategy and high-level design for the future Openshift Virtualization solution architecture.

Foundation Phase

The Foundation phase of the OpenShift Virtualization migration project is focused on demonstrating technical solution feasibility, mitigating risks, understanding platform infrastructure and operational requirements, and developing appropriate platform management and automation capabilities (i.e. the 'migration factory'). These foundational capabilities provide technical validation and feedback to facilitate the planning of large scale Openshift production platform management and VM workloads migration that will be implemented during the subsequent Expand phase.

The foundation phase consists of two stages:

- **Migration Pilot**
  Deploy and validate a pilot Openshift virtualization environment. Migrate test VM workloads to the Openshift virtualization certification environment and complete a suite of functional, performance and resilience tests to certify the pilot environment.

- **Production Build and Migration Factory**
  Develop the necessary capabilities to provision and manage large numbers of Openshift virtualization clusters, and build advanced Ansible automation solutions to orchestrate large scale VM migrations.

Expand Phase

The expand phase is when the large scale VM migration waves are planned and executed. The scaled migrations of VM workloads are typically organized into a series of '**migration waves**' (*batches of VM workloads*) to provide high-level management structure and oversight. Also, segmenting the migration into sequential migration waves enables **lessons learned and improvements** to be incorporated into the migration factory thereby accelerating and enhancing subsequent migration waves. For this reason, it is recommended that relatively small batches be begun and the size of the migration wave be scaled progressively.

The implementation of large scale migration waves is made possible by leveraging the capabilities of Openshift production platform management and automated migration solutions that were developed in phase 1.2 (Production Build and Migration Factory).

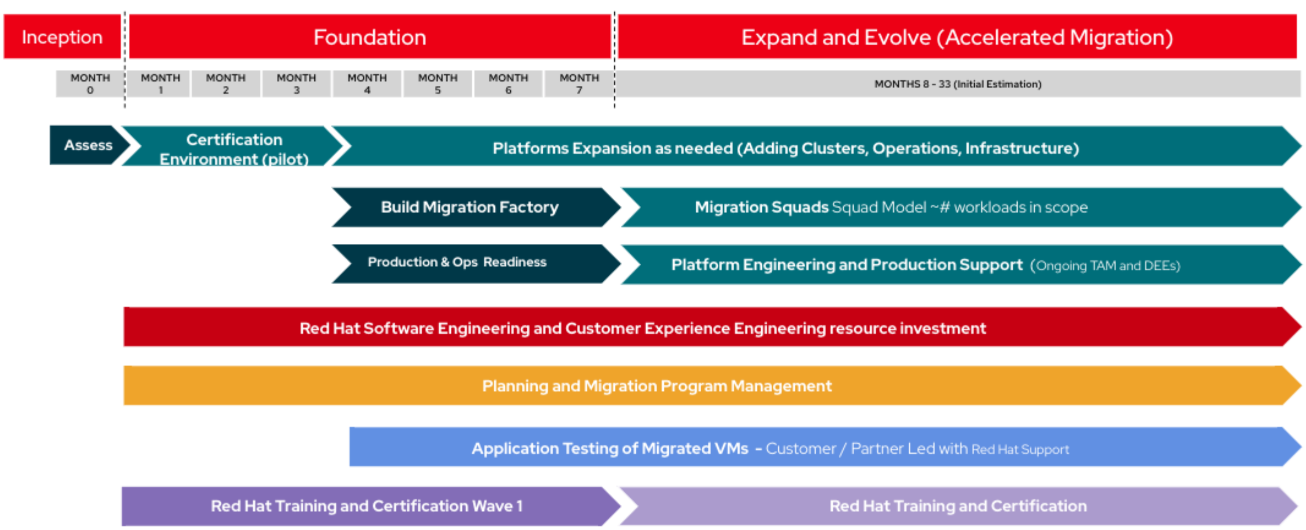
As noted previously, the migration schedule for DC and facility/distributed

- **Migrate at Scale**
  Define, plan and implement the migration of large numbers of VM workloads in a series of VM workload migration waves. Leverage the capabilities of ACM Openshift platform management and AAP automated migrations to implement large scale workload migration.

- **Optimize and Refine**
  Manage, track and monitor the outcomes and issues detected during a migration wave. Analyze issues found, develop solutions and incorporate improvements (technical and process) into the migration factory model to enhance the effectiveness of subsequent migrations waves.

Manufacturing Customer Migration Program Plan

The diagram below illustrates an indicative timeline for the Manufacturing Customer virtualization migration project. As can be seen, the project plan builds on the phased approach of virtualization migration project model defined above.



Manufacturing Customer VM Migration Project - Draft Timeline

Phase 0: Inception (complete)

Virtualization Migration Assessment (complete)

The Virtualization Migration Assessment ensures that Red Hat's knowledge of OpenShift, Openshift Virtualization, Migration Toolkit for Virtualization (MTV) and other technologies are properly communicated and demonstrated with Manufacturing Customer. Manufacturing Customer will ensure that enterprise knowledge of the environment, their requirements, and goals are properly communicated.

Tasks

- Manufacturing Customer provided an in-depth review of their current environment, problems with their current environment, and their desired future environment.
- Red Hat demoed Openshift, Openshift Virtualization, and the Migration Toolkit for Virtualization.
- Red Hat provided an in-depth review of the relevant technology and tools and proposed a HLD to meet Manufacturing Customer's virtualization migration goals.

Outputs

- High Level Design (HLD) for proposed OpenShift Virtualization solution
- Complexity Analysis of the current VM landscape
- Definition of the Pilot phase and scoped use-cases
- Certification Environment requirements
- Proposed Architecture and Bill of Materials for Pilot Phase

Phase 1: Foundation

Phase 1.1 - Virtualization Proof of Value Pilot (Certification Environment)

Phase Overview

The pilot lays the pathway for migration by preparing OpenShift infrastructure and related automation, defining and validating a strategy for migration, and developing procedures and providing training.

Based on the assessment findings and the joint identification of specific use-cases for the pilot phase, the following workstreams and tasks have been identified (refer the use cases listed in Proof of Value Pilot: Use Cases):

Pilot Workstreams

- **DC Virtualization workstream**: This workstream will focus on the use-cases identified for the DC deployment and VMs
- **facility/distributed workstream**: This workstream will focus on the use cases and unique scenarios specific to the facility/distributed scenarios
- **OpenShift infrastructure and automation workstream**: this workstream will enable the OpenShift Virtualization infrastructure deployment and the related automation (*including reuse of existing assets at Manufacturing Customer*)

Pilot Tasks

- OpenShift Infrastructure and Automation
  - Infrastructure low-level design
  - IaC / GitOps and AAP automation for infrastructure
  - OpenShift deployments
  - Process and automation for host recommissioning
- Pilot VM Migrations (with MTV)
  - Workload information gathering and evaluation
  - Process / strategy identification
  - Manual migration validation with real workloads
- Platform Operationalization (observability)
  - Metrics
  - Logging
  - Alarming
  - Operations team training / enablement
- Testing and Validation
  - Functional
  - Resilience
  - Performance

Outputs

- One Pilot environment (carried over from Certification Environment)
- Non-production environment
- Initial VM migrations and pilot use-cases tested with pilot workloads

Phase 1.2 - Production Build & Migration Factory

Phase Overview

Implementing a large-scale virtualization migration project requires the ability to deploy, manage and monitor large numbers of Openshift Virtualization clusters in full compliance with Manufacturing Customer's production systems requirements. These production platform management capabilities are developed during this phase in the *Openshift Virtualization Production and Ops Readiness* workstream.

In addition to large-sale openshift platform management capabilities, the task of migrating thousands of VMs requires an advanced automation solution to implement and validate the migration of large batches of VMs from the source to the destination platforms. This high level automation / migration orchestration capabilities are developed in the *Virtualization Migration Factory* workstream. In addition, since Manufacturing Customer has a mature set of automation and GitOps and IaC based artifacts and patterns already established for its existing OpenShift based CaaS clusters, this provides an opportunity to reuse these assets to the extent possible for the build and operations of the new OpenShift Virtualization clusters.

Finally, as part of this phase it is also important to review and strategize on the most effective and efficient approach to migrate large numbers of VM workloads to the target platforms. These activities are carried out as part of the *Prepare Scaled Migration Approach* workstream.

Various organizational, procedural and technical factors can influence migration strategy. In general two major styles of migration factory approaches can be followed: infrastructure team-led migrations and self-service migrations. From a capability development and risk management perspective it often makes sense to begin with infrastructure team-led migrations and evolve to self-service migrations as the migration process matures.

Workstreams & Tasks

- **Openshift Virtualization Production and Ops Readiness**
  - Design and deployment of the ACM Hub Openshift cluster (Management cluster)
  - Openshift Virtualization cluster provisioning and management, and day-2 Ops with ACM
  - Integration with Manufacturing Customer Monitoring and Alerting solution(s)
  - Integration with Manufacturing Customer Marketplace

- **Virtualization Migration Factory**
  - Design and deployment of Ansible Automation Platform (Automation Hub cluster) for Migration factory
  - Design, implementation and validation of Ansible Automation playbooks for the following activities:
    - Batch VM Migration
    - Pre and Post migration configurations
    - Rollback and recovery
    - VM validation

- **Prepare Scaled Migration Approach**
  - Strategize migration approach (Infrastructure led, and/or Self service)
  - Migration Self Service capabilities
  - Migration Planning: Identify related Workloads such as Clusters, 3-tier applications, dependent microservices.
  - Manual migration for exceptions strategy
  - Acceptable Outage duration and migration window
  - Notification List
  - Preflight Checks
  - Success Criteria

Outputs

- ACM deployed and Validated
- AAP deployed and Validated
- Initial migration factory content developed and validated
- Am initial migration iteration/wave executed
- Metrics Collected
  - Speed (migration velocity)
  - Errors
  - Outage times
- Identified opportunities for improvement

Milestone Estimates

This estimation is focused on migrating virtual machines as quickly as practical and leaving any application modernization opportunities for later. And the dates are estimates. The accuracy and confidence of the estimates depend on the result of the Pilot and workforce assigned.

Assumptions

Due to the limited discovery at this point in the effort, these estimates are based on assumptions outlined below:

- 13,735 Virtual Machines are in scope
- CDC: No VM or network information for the site was provided, hence it is not included in this estimate.
- Approximately four (4) months is planned for performing the Proof of Value Migration Pilot
- All security approvals to operate the platform will have been obtained during the Pilot phase
- Pilot workloads will consist of a cross-section of easy, medium, and hard workloads
- Change management will be integrated with the migration process through a “standard change” or similar process that will allow migration of virtual machines at will, throughout the workday, after hours, and on weekends provided all migration automation is signed off by Manufacturing Customer.
- The supported migration back-out procedure consisting of powering the source VM back on and deleting the migrated VM (losing any changed disk state) is acceptable for at least 60% of VM workloads.
- VDI is out of the scope of this migration effort.
- Virtual appliance lists were not provided at time of assessment and will need to be provided for refining the migration plan.

- Manufacturing Customer will need to provide detailed information on all planned freezes and downtimes for DC's and facility locations to enable accurate planning of migration efforts to avoid any potential impacts to Manufacturing Customer's working environment.
- Maintenance windows will be scheduled for appropriate periods of time, especially when migrating large images. Maintenance windows will be created for the DC's and facility environments outside of the existing limited schedule based on the agreed migration schedule.

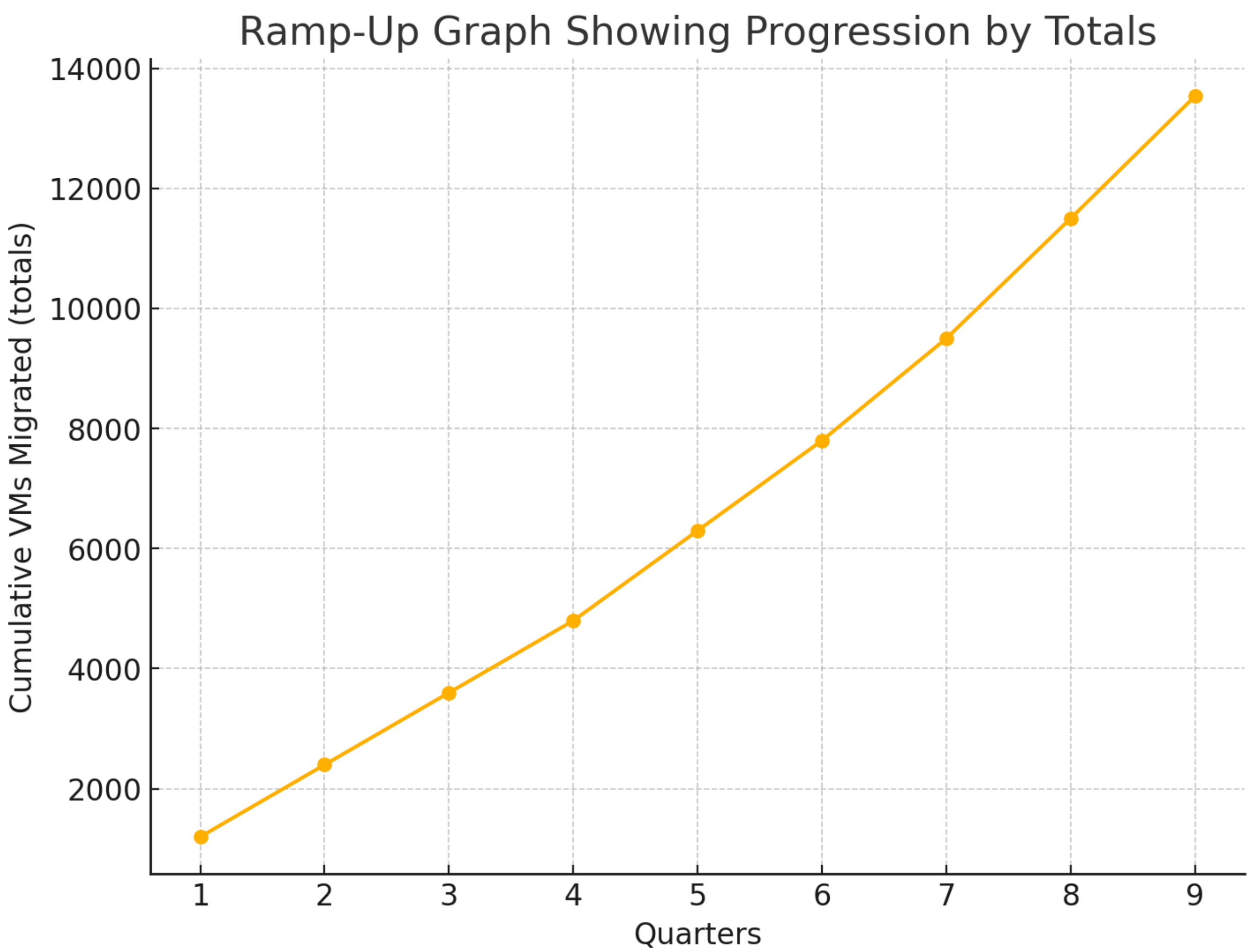
Based on the assumptions above and the complexity analysis, Red Hat proposes the following indicative schedule for the migration of the in-scope VMs. Note that this is an initial estimate and will need to be refined based on the assumption resolution above, including the impact of release and change windows for both DCs and facility/distributed sites.

Indicative High-Level Migration Schedule (to be refined)

Date (MM/DD/YY)	VMs Migrated on Wave	VMs Retired	vSphere VMs Remaining
PILOT PHASE BEGINS			
10/01/20xx	0	0	13,735
PILOT PHASE ENDS / FIRST MIGRATION WAVE STARTS			
02/01/20xx	200	0	13,535
MIGRATION WAVES CONTINUE			
03/01/20xx	1,200	0	12,335
06/01/20xx	1,200	0	11,135
09/01/20xx	1,200	0	9,935
12/01/20xx	1,200	0	8,735
03/01/20xx	1,500	0	7,235
06/01/20xx	1,500	0	5,735
09/01/20xx	1,700	0	4,035
12/01/20xx	2,000	0	2,035
03/01/20xx	2,035	0	0
MIGRATION FACTORY ENDS / ONGOING SUPPORT BEGINS			

07/01/20xx	13,735	0	0

Based on the previous table, here is a graph that illustrates the migration process ramp-up period per quarter (3 months period).



Red Hat provides these dates based on estimates to Manufacturing Customer. Migration timelines could be affected by multiple factors outside of Red Hat's control, including, but not limited to:

- maintenance and change windows
- availability of application teams to perform data validation
- external integrations, e.g.: network, dns entries, ipam, storage, security groups, firewall rules
- hardware procurement delays
- data center dependencies including space, cooling, and power

Certification Environment for Pilot

Overview

Prior to beginning the migration, a Proof of Value (PoV) pilot in a certification environment will be implemented by Manufacturing Customer and Red Hat teams. This environment will prove the functionality, performance and reliability of the proposed cluster architecture as it would be implemented inside of Manufacturing Customer's network and data center constraints.

Target use cases for this pilot are listed in the Proof of Value Pilot: Use Cases section above. Finalization of these use cases for the above targets will be a required prerequisite to the pilot effort start and will be collaboratively defined by Red Hat and Manufacturing Customer.

Red Hat envisions at least three (3) separate pilot clusters in the certification environment:

1. A cluster for the DC use cases
2. A cluster for the facility/distributed use cases, and
3. An SNO cluster for proving the standalone use-case.

In addition, to complete the validation of the selected use-cases, Red Hat recommends that Manufacturing Customer identify a small set of candidate VMs to use for trial migrations during this pilot.

Hardware

The certification environment provided by Manufacturing Customer is based on HP server compute hardware.

Network

The existing hardware is repurposed, so each cluster node will have two network interfaces consisting of fast interfaces on non-data center 10G x4 (2 for storage, 2 for data), and data center some 100G x2, and some 10 or 25Gb x4. An LACP-based link aggregation configuration is recommended for the bonded interfaces.

Each server is also expected to have an OOB management interface which will be used for operating system provisioning (virtual media) and also for fencing to expedite workload recovery in the event of a node failure. If this OOB network is not reachable, then firewall rules will be needed to allow access from the management network to the OOB interfaces by HTTPS and vice-versa.

Networking requirements

Every OpenShift cluster deployment requires the following:

- VLAN for OpenShift Network (OCP VLAN).
- Addressing (CIDR) for nodes in the OCP VLAN.
- OpenShift SDN (OVN-Kubernetes) internal/private addressing (potentially reusable across all clusters):
    - Pods/Cluster Network.
    - Services Network.
- A round trip time of 10 milliseconds (RTT) between sites is required (between control-plane nodes)This requirement is for the etcds.

Each cluster will require configuration at the switching level to allow access to VLANs where VMs will operate. Since lift-and-shift migration of virtual machines using MTV is expected, these VLANs must be propagated in both the VMware and OCP-v environments. This allows for the reuse of the same IPs and MACs from the current VMs after migration to the new platform.

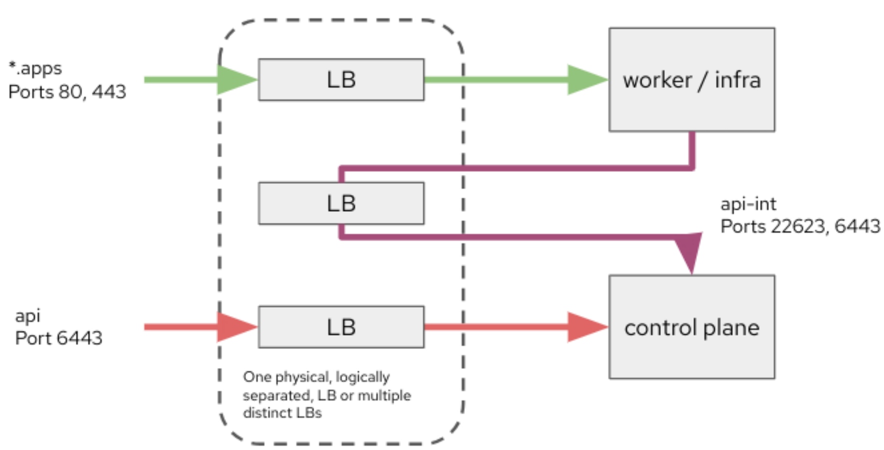
Load Balancer

OpenShift depends on load balancing control plane and containerized application traffic across a highly available set of controller nodes and ingress controllers, respectively. With baremetal clusters, this function can be provided by an internal VRRP-based mechanism or by a highly-available external load balancer such as an F5 LTM or Citrix Netscaler installation.

External load balancing is most appropriate for Manufacturing Customer's environment to provide a consistent configuration though internal load balancing may be used in those environments for testing purposes.

Every OpenShift cluster deployment will need and external Load Balancer L4 for:

- Internal API (api-int.)
- External API (api.)
- Ingress Controller (*.apps.)



DNS

Every cluster will generally need three cluster DNS records representing the three load balanced virtual servers.

- Internal API: *api-int.<cluster_name>.<base_domain>*
- External API: *api.<cluster_name>.<base_domain>*
- Ingress Controller: **.apps.<cluster_name>.<base_domain>*

another option would be to allow control to self manage the entire <ocp-subdomain>.<base_domain>, this greatly facilitates operations

These records can be created using dotted names within a single DNS zone or by creating a per-cluster DNS zone at the preference of Manufacturing Customer. The wildcard ingress DNS record can be replaced by static ingress DNS records if required; however this increases the management overhead for the cluster. More information about OpenShift's DNS requirements can be found in the official documentation.

Connectivity

- Network connectivity between the OpenShift pod network and the management interface (out-of-band) is crucial for several scenarios:
  - For automatic baremetal installations (IPI or Assisted Installer), this connectivity is mandatory.
  - If we plan to utilize the Fence Agents Remediation (FAR) for VM High Availability, the management interface must be reachable from the OpenShift pod network. Without this connectivity, we would need to employ the Self Node Remediation (SNR) operator, which ensures VM HA but with slower recovery times for Virtual Machines.
- Access to a container registry is essential for managing OpenShift cluster images, Operator images, and catalogs.
- Machine with internet access for mirroring container images, virtual machine OS images, and templates.
- NTP servers. If you intend to use a local enterprise NTP server or if your cluster is deployed in a disconnected network, you can configure the cluster to utilize a specific time server. Refer to the following KCS for further information.
- Access to code repositories allowing for GitOps deployments

Storage

Storage in the data centers is set up with failure domains to provide maximum resilience. NetApp provides the backing storage for ESXi LUNs over NFS v3 and provides backups leveraging volume snapshotting.

- Local storage is leveraged for ESXi hosts to boot. Currently a mix of flash drives, NVME is used. The differences arise from the purchase date, as the hardware used different storage media. There is no boot from SAN or SD card
- Clusters that are not using NetApp are using “localstorage”
- Fusion SDS is used in the current OpenShift (CaaS) environments
- Cohesity was discussed for incremental backups RedHat is actively discussing with Cohesity there is currently no ETA

Validation tests for certification environment

Functional test

In addition to the functional and capability validation use cases documented in the pilot section, Red Hat recommends performing additional testing using the following use cases:

Use cases

Red Hat

Num.	Name	Topic/description
1	Run VMs	Test if a VM can be created with storage provided by Fusion SDS and an interface attached to enable communication with other VMs outside the cluster.
2	Delete VMs	VMs can be deleted
3	Add new VLAN as new network	Use the NMstate operator to configure a new network. Attach an interface in the new network to a VM
4	Instance sizing	Able to create instances customizing nº cpus, memory, nº od disks, disk size, disk type (if available) and nº of NICs
5	Different OS	Templates with different OS, for example: different versions of RHEL, Windows 10/11, Windows Server, etc
6	Instance resizing	Instance resources can scale up/down automatically or manually without service disruption.
7	Share disk between instances	A disk can be shared to one or more instances with a cluster filesystem
8	Clone instance	Instances can be cloned without service disruption (1 or more disks)
9	Clone disk	Instance disks or standalone disk can be cloned without service disruption
10	Attach ISO images	ISO images can be connected/disconnected to instances without service disruption
11	Access to instance console	Instance local console can be access securely
12	Convert/clone instance into template	Convert/clone instance into template without service disruption
13	Connect or disconnect NICs	A NIC can be added and removed from instance without OS power off
14	Add/change/remove instance tags	instance has tags and these can be added, removed or changed without service disruption
15	Add/change/remove instance description	instance has description and this can be added, removed or changed without service disruption
16	Instance Snapshot	Instances can have snapshots taken/delete without service disruption
17	Instances affinity rules	instance groups can be defined along with affinity/ antiaffinity rules for resiliency/performance purposes

Num.	Name	Topic/description
18	Instance metrics	Metric about instance resource consumption (eg. cpu, mem, disk)
19	Instance HA	Testing of HA methods, fencing, and automatic live migration
20	Numa configuration	Instance configuration allow Numa config and allocation
21	VM migration	Migration tool cover migrate VMs from vsphere to Openshift Virt
22	Automated migration plan	Migration tool support full migration plan including previous syncs, sync frequency, and power off scheduling)
23	Move a VM between nodes, enclosures, racks and data centers.	move VM from one leaf to the next, see what ARP tables do
24	Migration tests	Data transfer directly from ESXi host
25	Migration tests	Data transfer through vsphere management environment
26	Cluster Upgrade	Test an upgrade of the cluster running VMs on top of Openshift without service disruption
27	Power off node with running VMs	Power off node and observe behavior of cluster

Large-scale solution

Hub Cluster (Management)

Red Hat Advanced Cluster Management for Kubernetes (ACM) provides end-to-end management visibility and control to manage your Kubernetes environment. Take control of your application modernisation program with management capabilities for cluster creation and application lifecycle and provide security and compliance across hybrid cloud environments. Clusters and applications are all visible and managed from a single console with built-in security policies. Run your operations from anywhere Red Hat OpenShift runs and manage any Kubernetes cluster in your fleet.

With Red Hat Advanced Cluster Management for Kubernetes:

- Work across various environments, including multiple data centers, private clouds and public clouds that run Kubernetes clusters.
- Easily create Kubernetes clusters and offer cluster lifecycle management in a single console.

- Enforce policies at the target clusters using Kubernetes-supported custom resource definitions.
- Deploy and maintain day-two operations of business applications distributed across your cluster landscape.

A Hub Cluster running ACM will help Openshift Virtualization mainly in three ways:

- The **deployment** and **management** of Openshift Clusters
- **Observability** of aggregated metrics data from OpenShift Clusters
- **Governance** and enforcement of configuration standards

The deployment of an ACM hub cluster will be crucial for the large-scale deployment and management of clusters and data centers.

At global scales, a single ACM hub cluster will be insufficient and regional or datacenter-level hub clusters will be needed. All hub clusters can be aggregated into a true global single-pane solution called **Global Hub**.

OpenShift Virtualization Recommendations

Openshift releases

When setting up your OpenShift environment, it is crucial to select the appropriate version to ensure stability, security, and access to the latest features. We always recommend using the latest version of OpenShift for several reasons:

- **Enhanced Security:** The latest version includes the most recent security patches and updates, protecting your environment from vulnerabilities.
- **New Features:** Each new release brings innovative features and improvements that enhance functionality and user experience.
- **Performance Improvements:** Up-to-date versions often come with optimizations and performance enhancements, ensuring your system runs efficiently.
- **Support and Compatibility:** The latest version ensures compatibility with the newest tools and technologies, and typically receives the most active support from Red Hat.

For the certification environment and proof of value phase, we will use Openshift 4.16+. In subsequent phases, the version to be used will be reviewed.

Installing disconnected environments

This architecture assumes usage of the installer provisioned infrastructure (IPI) method for deploying the cluster. The resulting cluster will have the bare metal cloud provider configured and will be able to make use of it for adding and managing nodes. For more information on IPI-based bare metal installations, please see the documentation here.

For the suggested cluster designs in this reference architecture, the control plane should be schedulable. When instantiating the cluster, deploy a three-node “compact” cluster, which will mark the control plane nodes as schedulable by default. After deployment, join additional compute and storage nodes as needed, using distinct MachineSets for each role, e.g. storage and compute.

Red Hat Enterprise Linux CoreOS (RHCOS)

OpenShift Virtualization requires Red Hat Enterprise Linux CoreOS (RHCOS) compute nodes. Even though it is possible to deploy and use Red Hat Enterprise Linux (RHEL) compute nodes, they are incompatible with OpenShift Virtualization. RHCOS is based on RHEL and designed to run containers with an integrated CRI-O runtime and dedicated container tools. The operating system uses rpmostree to facilitate being managed through the Machine Config Operator to deploy operating system updates and apply configuration declaratively.

To customize RHCOS, for example, to add drivers or other third-party OS-level tools, the RHCOS image layering feature must be used to modify the operating system and add the appropriate packages.

For these compelling reasons, OpenShift Virtualization must be deployed on nodes running RHCOS.

For more information on Red Hat Enterprise Linux CoreOS, see the documentation.

Post-install configuration

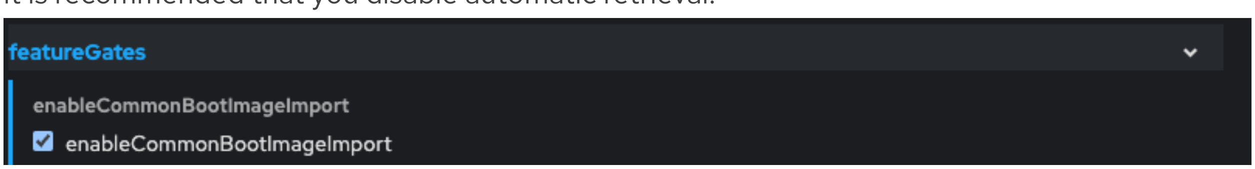
After the OpenShift cluster has been deployed, with additional nodes joined for compute roles, additional cluster configuration is needed to adjust and optimize the configuration for virtual machines.

OpenShift Virtualization Operator Configuration

OpenShift Virtualization is deployed and configured in the cluster using OperatorHub. Once installed, there are several configuration options to set. This means the configuration must point to the registry at Manufacturing Customer to provide access to the Operators. This also means the registry must mirror the latest versions of the required Operators.

Follow the documentation for deploying the Operator and creating an instance of the HyperConverged resource. The below bullets represent common decisions and configuration parameters that are not default.

- Decide whether to download the default, Red Hat provided operating system images. The Operator will automatically try to retrieve and make available cloud images for RHEL, Fedora, and CentOS when configuring a default storage class. If you do not intend to use these images, it is recommended that you disable automatic retrieval.



- Configure the default CPU model to the lowest common denominator CPU model in the cluster. These values are determined by the CPU vendor and the highest or lowest, CPU model available in the OpenShift cluster. For AMD CPUs, a value of EPYC is recommended; for Intel

CPUs a value equivalent to the generation name, e.g. Cascadelake-Server, is recommended.

defaultCPUModel

```
Cascadelake-Server
```

The Intel Xeon-Gold 6258R CPU is of the Cascadelake-Server type.

- Set the VM run strategy to use the RerunOnFailure policy.

- If you do not want to use the default storage class for storing VM state related to, among other things, vTPM, set the desired storage class when deploying OpenShift Virtualization. This storage class must support RWX access mode.

vmStateStorageClass

```
ocs-storagecluster-ceph-rbd-virtualization
```

Machine and node configuration

These tasks represent configurations applied to the cluster nodes after the cluster is deployed.

- Configure time synchronization to avoid clock drift and issues related to inaccurate/inconsistent time, such as certificates being incorrectly marked invalid.

- Configure node health checks and fence agent remediation for reduced HA recovery time in the event of node failure.

- The Node Health Check Operator and Fence Agents Remediation Operator are deployed from OperatorHub in the web console. Once installed, an instance of the FenceAgentsRemediationTemplate must be created to provide BMC connection information for the nodes. This is the same information that was used when deploying the cluster and joining nodes using a MachineSet. Finally, create an instance of the NodeHealthCheck to define appropriate timeout values for your desired VM recovery time. Refer to the chart in this KCS for details on how different values affect the recovery times.

- The Kube Descheduler Operator is used to evict workloads when load across the cluster is not rightly distributed.

- The Numa Resources Operator is integrated with the Node Tuning Operator and can be implemented to support scheduling NUMA-aware workloads.

Kubelet configuration

The following additional configuration for kubelet is applied after the cluster is deployed.

- Increase kubelet kubeAPIBurst to 200 and kubeAPIQPS to 100.
  Adjusting these values up from the default of 100 and 50, respectively, accommodates bulk object creation on the nodes. The lower values are useful on clusters with smaller nodes to keep API server resource utilization reasonable, however, with larger nodes, this is not an issue.

- Set the maxPods per node to 500. By default, OpenShift sets the maximum Pods per node to 250. This value affects not just the Pods running core OpenShift services and node functions but also virtual machines. This value is likely too small for large virtualization nodes that can host many virtual machines. If you're using very large nodes and may have more than 500 VMs and Pods on a node, this value can be increased beyond 500, however, you will also need to adjust the size of the cluster network's host prefix when deploying the cluster.

- Disable nodeStatusMaxImage. The scheduler factors both the count of container images and which container images are on a host when deciding where to place a Pod or virtual machine. For large nodes with many different Pods and VMs, this can lead to unnecessary and undesired behavior. Disabling the imageLocality scheduler plugin by setting nodeStatusMaxImage to -1 facilitates balanced scheduling across cluster nodes, avoiding scenarios where VMs are scheduled to the same host as a result of the image already being present vs factoring in other resource availability.

- Set dynamic resource allocation for kubelet. The default CPU and memory reservation for kubelet is very small and not appropriate for nodes with large amounts of resources, such as hypervisor hosts. Setting dynamic resource allocation will set the values for system reserved CPU and memory according to the total amount of resources on the node. This prevents kubelet from starving for resources when the node has high numbers of Pods and virtual machines running.

- Configure soft eviction thresholds. Configuring soft eviction is valuable for several reasons, however the most important is that it sets the upper boundary for memory utilization on the nodes before the virtual machines are attempted to be moved to other hosts in the cluster. This value should be set for a reasonable value according to the approximate maximum amount of memory utilization you want on the node; however, if all nodes exceed this value, then the workload will not be rescheduled. Depending on the amount of memory in your hosts, this could be between 90-95%. For the suggested node size in this architecture, a value of 90% is used.

  Additional eviction thresholds for local storage utilization can be configured based on needs, in particular, if the disks used by RHCOS are smaller (less than 512GiB) or you intend to deploy ephemeral virtual machines with disks stored in container images.

Below is an example kubelet configuration to apply the above suggested configuration to compute nodes in the cluster. If you have hypervisor hosts that do not have the “worker” label, an additional selector may be needed.

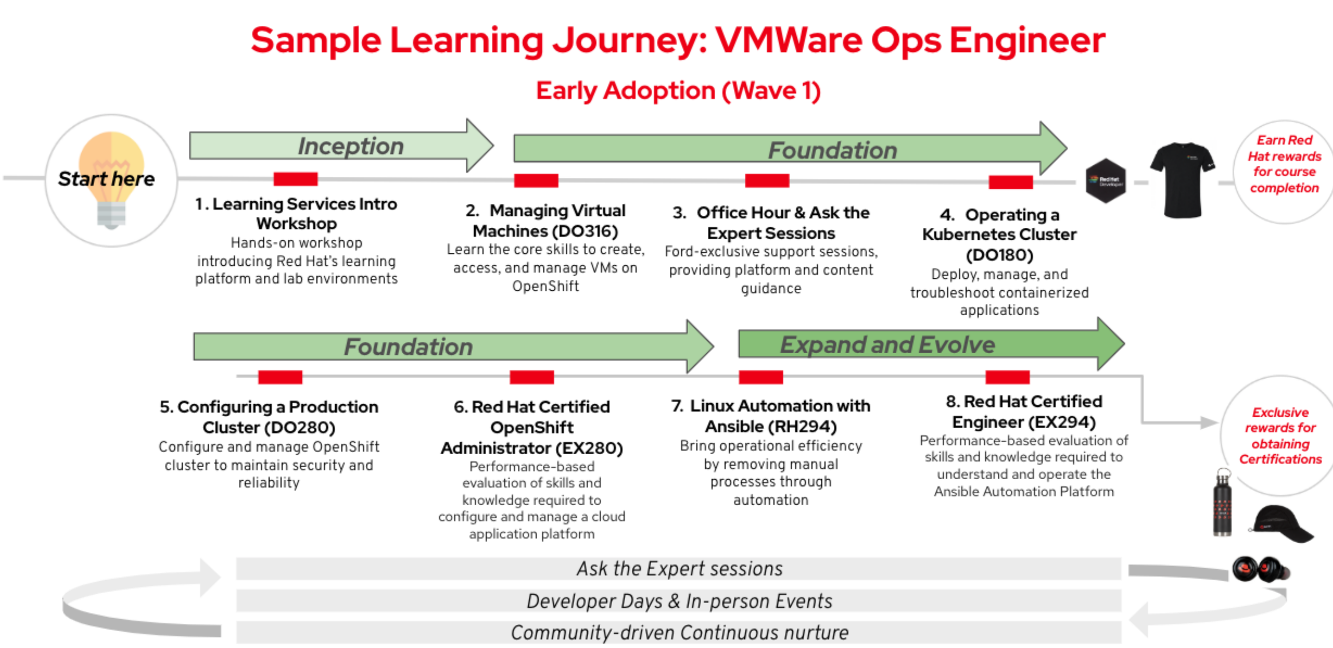
```
apiVersion: machineconfiguration.openshift.io/v1
kind: KubeletConfig
metadata:
  name: set-virt-values
spec:
  machineConfigPoolSelector:
    matchLabels:
      pools.operator.machineconfiguration.openshift.io/worker: ""
  kubeletConfig:
    autoSizingReserved: true
    maxPods: 500
    nodeStatusMaxImages: -1
    kubeAPIBurst: 200
    kubeAPIQPS: 100
    evictionSoft:
      memory.available: "50Gi"
    evictionSoftGracePeriod:
      memory.available: "5m"
    evictionPressureTransitionPeriod: 0s
```

Enablement and training

Training and Enablement are critical elements of a successful migration to the Red Hat OpenShift Virtualization Platform. Red Hat has created a prescriptive and tailored training strategy for Manufacturing Customer, based on the following pillars:

- **Role-Based Learning Paths**: Course curriculums and certifications, created according to each team's unique goals and personas.
- **Enablement Timeline**: Mapping the learning journey to the various stages of Manufacturing Customer's migration.
- **Tailored Course Content**: Breakdown of the core foundation learning, by course, with expected outcomes and user goals.
- **Flexible Delivery Methods**: Hybrid learning approach which encompasses customized instructor-led training, as well as self-paced, to ensure successful long-term adoption of Manufacturing Customer's ongoing digital transformation.

An example of a persona-based learning path (for a VMWare Ops Engineer transitioning to OpenShift Virtualization) is depicted below:



The following subsections provide a list of recommended Red Hat training and a high-level overview of each course.

DO180: Red Hat OpenShift Administration I: Operating a Production Cluster

DO180 prepares OpenShift cluster administrators to manage Kubernetes workloads and to collaborate with developers, DevOps engineers, system administrators, and SREs to ensure the availability of application workloads. This course focuses on managing typical end-user applications that are often accessible from a web or mobile UI and represent most cloud-native and containerized workloads. Managing applications also includes deploying and updating their dependencies, such as databases, messaging, and authentication systems.

The skills that you learn in this course apply to all versions of OpenShift, including Red Hat OpenShift on AWS (ROSA), Azure Red Hat OpenShift, and OpenShift.

Course Content Summary

- Managing OpenShift clusters from the command-line interface and the web console
- Deploying applications on OpenShift from container images, templates, and Kubernetes manifests
- Troubleshooting network connectivity between applications inside and outside an OpenShift cluster
- Connecting Kubernetes workloads to storage for application data
- Configuring Kubernetes workloads for high availability and reliability
- Managing updates to container images, settings, and Kubernetes manifests of an application

DO280: Red Hat OpenShift Administration II: Configuring a Production Cluster

DO280 prepares OpenShift Cluster Administrators to perform daily administration tasks on clusters that host applications provided by internal teams and external vendors, enable self-service for cluster users with different roles, and deploy applications that require special permissions such as such as CI/CD tooling, performance monitoring, and security scanners. This course focuses on configuring multi-tenancy and security features of OpenShift as well as managing OpenShift add-ons based on operators.

The skills you learn in this course can be applied using all versions of OpenShift, including Red Hat OpenShift on AWS (ROSA), Azure Red Hat OpenShift, and OpenShift.

Course Content Summary

- Deploying packaged applications using manifests, templates, kustomize, and helm.

- Configuring authentication and authorization for users and applications.
- Protecting network traffic with network policies and exposing applications with proper network access.
- Deploying and managing applications using resources manifests.
- Enabling developer self-service of application projects.
- Managing OpenShift cluster updates and Kubernetes operator updates.

DO316: Managing Virtual Machines with Red Hat OpenShift Virtualization

DO316 teaches the essential skills required to create and manage virtual machines (VM) on OpenShift using the Red Hat OpenShift Virtualization operator.

This course provides:

- Skills required to create, access, and manage VMs on OpenShift clusters.
- Skills required to control usage and access of cpu, memory, storage, and networking resources from VMs using the same Kubernetes features that would also control usage and access to these resources for containers.
- Sample architectures to manage High Availability (HA) of VMs using standard Kubernetes features and extensions from OpenShift Virtualization.
- Strategies to connect VMs on OpenShift to data center services outside of their OpenShift cluster, such as storage and databases.

Course Content Summary

- Create VMs from installation media and disk images.
- Access text and graphical consoles of a VM.
- Connect to VMs using Kubernetes networking (services, ingress, and routes)
- Provision storage to VMs using Kubernetes storage (PVC, PV, and storage classes).
- Start, pause, and stop VMs.
- Clone and snapshot VMs.
- Connect VMs to external and extra networks (outside of the Kubernetes pod and service networks).
- Connect VMs to host storage and external storage.
- Ansible management of VMs.
- Create VMs from VM Templates.

DO370: Enterprise Kubernetes Storage with Red Hat OpenShift Data Foundation

Teaches the essential skills required to design, implement, and manage a Red Hat OpenShift Data Foundation cluster and perform day-to-day Kubernetes storage management tasks.

Traditional storage options available to Kubernetes administrators are limited and lack flexibility and/or versatility. Red Hat OpenShift Data Foundation provides real advantages, even when it is backed by cloud storage such as AWS EBS and sophisticated on-prem legacy storage like SAN arrays. Many companies rely on third-party solutions to manage backup and disaster recovery in production. However, proper planning to implement these solutions requires knowledge of the Kubernetes CSI and OAPD APIs. This course walks the student through the recommended steps of configuring and managing storage services for container and Kubernetes services.

Course content summary

- Deploy Red Hat OpenShift Data Foundation in internal and external mode.
- Provision non-shareable block storage to applications like databases.
- Provision shareable block storage to applications like virtual machines.
- Provision shareable file storage to such applications as CI/CD pipelines and AI/ML.
- Provision shareable object storage to applications, such as AI/ML and media streaming.
- Provision storage for Red Hat OpenShift cluster services, such as monitoring and registry.
- Monitor and expand storage capacity and performance
- Attach and detach storage from an application for backup and archiving.
- Create and access volume snapshots and clones.
- Troubleshoot internal Ceph components of Red Hat OpenShift Data Foundation.
- Perform backup and restore operations using the OADP API.

DO380: Red Hat OpenShift Administration III: Scaling Deployments in the Enterprise

DO380 expands upon the skills required to plan, implement, and manage OpenShift® clusters in the enterprise. You will learn how to configure and manage OpenShift clusters at scale to address increasing and special demands from applications and ensure reliability, performance, and availability.

Course Content Summary

- Manage OpenShift cluster operators and add operators.
- Implement GitOps workflows using OpenShift GitOps operator.
- Integrate OpenShift with enterprise authentication.
- Query and visualize cluster-wide logs, metrics, and alerts.
- Backup and restore application settings and data with OpenShift APIs for Data Protection (OADP).
- Manage machine pools and machine configurations.

DO480: Multicluster Management with Red Hat OpenShift Platform Plus

DO480 teaches the skills required to maintain a diverse portfolio of applications running across a fleet of OpenShift clusters. Applications follow placement rules determined by capacity and criticality; cluster configurations comply with governance and security policies; all automated according to DevOps principles.

Course Content Summary

- Deploy Red Hat Advanced Cluster Management for Kubernetes (RHACM) in a hub cluster.
- Add a managed cluster to RHACM (configure a cluster to be managed by RHACM)
- Define and apply cluster configuration policies
- Detect and correct non-conformance to cluster configuration policies
- Visualize and compare cluster settings between different clusters
- Define and apply application placement policies
- Identify and compare application resources from multiple clusters
- Deploy Red Hat Quay in the hub cluster
- Deploy Red Hat Advanced Cluster Security for Kubernetes (RHACS) in the hub cluster.
- Integrate Red Hat Quay and RHACS with RHACM.

Appendix A: Supported Guest Operating Systems

Here is the master link for the Red Hat maintained supportability list: https://access.redhat.com/articles/973163?extIdCarryOver=true&sc_cid=701f2000001Css5AAC#ocpvirt

The following is the list of Red Hat and third-party guest operating systems that are certified and supported for use with Red Hat OpenShift Virtualization hosts.

Guest		OpenShift Virtualization version				Support Tier
Operating System	**OS Architecture**	**4.12**	**4.13**	**4.14**	**4.15**	
Red Hat Enterprise Linux 6	64 bit (x86)	✔	Not supported due to EOL [1]	Not supported due to EOL [1]	Not supported due to EOL [1]	Tier 1
Red Hat Enterprise Linux 7	64 bit (x86)	✔	✔	✔	✔	Tier 1
Red Hat Enterprise Linux 8	64 bit (x86)	✔	✔	✔	✔	Tier 1
Red Hat Enterprise Linux 9	64 bit (x86)	✔	✔	✔	✔	Tier 1
Microsoft Windows 10	64 bit (x86)	✔	✔	✔	✔	Tier 1

Guest		OpenShift Virtualization version				Support Tier
Microsoft Windows 11	64 bit (x86)	Not Supported	Not Supported	✓	✓	Tier 1
Microsoft Windows Server 2012 R2	64 bit (x86)	✓	✓	Not supported due to EOL [1]	Not supported due to EOL [1]	Tier 1
Microsoft Windows Server 2016	64 bit (x86)	✓	✓	✓	✓	Tier 1
Microsoft Windows Server 2019	64 bit (x86)	✓	✓	✓	✓	Tier 1
Microsoft Windows Server 2022	64 bit (x86)	Not Supported	Not Supported	✓	✓	Tier 1

1. The operating system has reached its end of life. Customers are recommended to review with the guest operating system vendor support policies relating to support for end of life products.

In addition to supporting Microsoft Windows operating systems, OpenShift Virtualization supports the following features for specific versions of Microsoft Windows :

Windows	10/11	Server 2012r2 / 2016 / 2019	Server 2022
WSL2	4.14 or later	No	4.14 or later
vTPM	4.14 or later	No	4.14 or later

SecureBoot	4.14 or later	No	4.14 or later
Nesting (vmx/svm)	No*	No*	No*

- Nested virtualization is currently offered as a Technology preview: https://access.redhat.com/solutions/6692341

- Other operating systems, like non-current RHEL and Windows versions or other OS like Debian or Ubuntu are categorized as “third party components” and are supported following Red Hat’s “Support Policy regarding third-party components” described in the following article.

Appendix B: Recommended Bill of Materials

Existing Environment

Infrastructure

Component	Vendor
Server Hardware Model	HP ProLiant DL320 G11, DL 360 G10+, DL380 G9/G10/G11, DL560 G10, Synergy 480 G10
SAN (Storage Area Network)	NetApp OnTap 9.10 / 9.13p6(?)
SDS (Software Defined Storage)	CaaS: PortWorx and Fusion SDS
File Storage	NetApp with NFS (VMware), NetApp iSCSI for Fusion SDS
Object Storage	Google Cloud Storage, S3 Compatible on NetApp Storage Grid (only in China)
Networking: Switching	Cisco Nexus 9K
Networking: Routing	Cisco facilities going to very (ACI in DC)
Networking: Firewall	F5
Networking: Load-Balancer	F5 LTM/DNS (aka GTM)
SDN (Software Defined Network)	NSX on VMware, OVN on CaaS, ACI Fabric (DCs)
VDI (Virtual Desktop Infrastructure)	VDI - different strategy / out of scope (Citrix)

VMware Portfolio

VMware Subscription Level	Select	Subscription / Core Count
VMware Cloud Foundation (VCF)		
VMware vSphere Foundation (VVF)	X	
VMware vSphere Standard (VVS)		
VMware vSphere Essentials Plus Kit (VVEP)		

VMware Product	Feature	Used (Y/N)
vSphere		Y
vSAN	Supported Storage Protocol	N
	Dynamic Volume Provisioning	N
	RWX for Live Migration	N
	Snapshot Support	N
	Clone Support	N
NSX (small set of clusters)	Microsegmentation	N
	Multi-Cloud Networking	Y
	Tunnels (IPSec, VPN)	Y
	Dynamic Routing (Distributed / Logical)	Y
	Central Network Management	N
	MPLS	N
	QoS	N
Aria Operations (vROP)	Performance Analytics (vSphere VMs)	Y
	Health Score	Y

VMware Product	Feature	Used (Y/N)
	Alerting	Y
BMC (today)	ITOM/ITSM integration (ServiceNow)	Y
	Recommendation Engine (e.g. rightsizing)	Y
	Automated Optimization	N
	True Visibility Suite	Y
Aria Automation (vRA)/vRo	Infrastructure provisioning (LCM)	Y
	Application Blueprints/workflows	Y
(Chef/Habitat)	Configuration Management	Y
	Service Catalog	Y
	Cloud Assembly	Y
Aria Log Insights		Y
HCX (Hybrid Cloud Extensions)		N
DSM (Data Services Manager)		N
DRS (Distributed Resource Scheduler)		Y
Storage DRS	use when needed	N
VDS (vSphere Distributed Switch)		Y
VMware Site Recovery Manager		Y
VMware Tanzu Kubernetes Service		N
VMware Tanzu Application Service (TAS, PCF)		N

3rd Party ISV Portfolio

Component	Existing Vendor
Backup & Recovery	NetApp Snap, SQL Veeam, IBM spectrum protect (TSM) / Tape *facilities, Cohesity (future)
DR	SRM, MSSQL Always ON, Oracle-Mirror, IBM spectrum protect (TSM) / Tape *facilities,NetApp Snap, SQL Veeam, ZDLRA
Monitoring	Dynatrace, WhatsUp Gold, vRealize, ACM Observability, Coming NetApp Insights,j
Logging	Splunk, Google Bucket, VM Insights, QRadar,
Metrics Collection & Alerts	Alertmanager, Webex Teams workplace, BMC, vRealize, AI-Ops , Turbonomic
Secrets Management	HashiCorp Vault
Certificate Management	Globalsign
Security in VM/Container	AV - ACS (Container Scanning) ,Cisco Traffic Watch and ACLs, vTPM, data encrypted at rest (SAN/NAS), compliance operator
Day 1 Operations	Habitat,chef,custom scripts
Automation and Configuration Management	Habitat,chef,custom scripts

Recommended Red Hat Environment

Red Hat Portfolio

Product	Quantity
OpenShift Virtualization Engine (Bare Metal Node) (1-2 sockets)	748
Advanced Cluster Management (1-2 sockets)	848
Ansible Automation Platform (100 Managed Nodes)	9
OpenShift Platform Plus (Bare Metal Node) (1-2 sockets)	100

Host Count per Category

Product	Quantity
# of hosts for Virtualization use case	600
# of hosts for Container Management use case	148
# of hosts for Application Platform use case	100

3rd Party ISV Portfolio

Component	Red Hat Recommended Vendor	Recommended Subscription
Backup & Recovery	Cohesity	TBD
DR	Cohesity	TBD
Monitoring	Current Solution: WhatsUP Gold,Cisco Workload monitoring, Dynatrace Add: ACM Observability, Alertmanager for Virt clusters	As-is + TBD
Logging	Current Solution: Splunk/ Google Bucket/shell script/ vRealize Log Insight Future: Logging 6.0 for OTEL	As-is + TBD
Metrics Collection & Alerts	Current Solution: WhatsUP Gold,Cisco Workload monitoring, Add: ACM Observability, Alertmanager for Virt clusters	As-is + TBD
Secrets Management	Current Solution: HashiCorp Vault	As-is
Certificate Management	Current Solution: Globalsign	As-Is
Security in VM/Container	Current Solution: AV - ACS (Container Scanning) ,Cisco Traffic Watch and ACLs, vTPM, data encrypted at rest (SAN/NAS), compliance operator	As-is
Day 1 Operations	Habitat,chef,custom scripts	Ansible Automation Platform (AAP)
Automation and Configuration Management	Habitat,chef,custom scripts	Ansible Automation Platform (AAP)

Appendix C: Glossary

Term	Definition
AAP	Ansible Automation Platform
ACM	Red Hat Advanced Cluster Manager
CBT	Change Block Tracking
DR	Disaster Recovery
HCP	Hosted Control Plane
HLD	High-Level Design
KVM	Kernel-based Virtual Machine
LLD	Low-Level Design
MTV	Migration Toolkit for Virtualization Operator
NVMe	Non-Volatile Memory Express
OVN	Open Virtual Network
OVS	Open Virtual Switch
PoV	Proof of Value
RHEL	Red Hat Enterprise Linux
RVTools	A free third-party utility that connects to the VMware vCenter server and captures information about virtual machines and ESXi hosts: CPU, Memory, Disks, Partitions, Networks, etc. (https://www.robware.net/home)
SRM	VMWare Site Recovery Manager
SVVP	Microsoft Server Virtualization Validation Program
V2C	Virtual to Container
V2P	Virtual to Physical
V2V	Virtual to Virtual
VDI	Virtual Desktop Infrastructure
VM	Virtual Machine