

VM infrastructure efficiency

Get more from your VM estate with Red Hat OpenShift Virtualization

IT leaders can use this structured, 4-stage approach (assess, recover, consolidate, modernize) to analyze how their organization uses its virtual machine (VM) estate. Each step of the checklist produces outcomes that can help extend the value of existing infrastructure investments while managing hardware and licensing risk.

1 Assess what you have

Outcome: A quantified view of utilization, waste, and dependencies. Define success in terms of recovered capacity, improved utilization, reduced platform duplication, cost avoidance, and operational readiness.

- Audit current VM utilization across your estate.** Identify idle, oversized, and underused VMs.
- Quantify wasted capacity in CPU, memory, and storage.** Determine what resources are allocated but not consumed.
- Benchmark your current per-VM infrastructure cost.** Establish a baseline before optimization.
- Review virtualization licensing exposure.** Model renewal costs under current and alternative terms.
- Identify who owns each workload.** Accountability is a prerequisite for rightsizing decisions.

2 Recover capacity from what you already own

Outcome: Measurable reduction in per-VM cost and infrastructure waste while still using existing hardware. This is continuous work.

- Rightsize VMs to actual workload requirements.** Apply rightsizing recommendations from Red Hat® Advanced Cluster Management for Kubernetes to match allocation to actual consumption.

- Reclaim resources from idle and orphaned VMs.** Decommission or consolidate where possible.
- Apply memory overcommitment where appropriate.** Start at 25% and tune based on workload behavior.
- Set a density target that reflects acceptable risk.** Density is a business decision, not a maximum to chase.

3 Consolidate where it makes sense

Outcome: A practical plan to reduce parallel infrastructure, improve utilization, and place workloads on the right platform without forcing application change.

- Identify where platforms are duplicated.** Look for environments where VM infrastructure and container platforms run side by side with separate hosts, tools, and operating models.
- Choose the right landing zone for each workload.** Decide which workloads should remain on-premise, run at the edge, or move to a supported public cloud. Choose a platform like Red Hat OpenShift® Virtualization that provides consistency across all 3.
- Use Red Hat migration tooling where VM movement is needed.** Use Red Hat OpenShift migration advisor for readiness planning and Red Hat's migration toolkit for virtualization to move suitable VM workloads to Red Hat OpenShift Virtualization.

- Sequence consolidation deliberately.** Start with lower-risk workloads to validate performance, operations, and capacity assumptions before moving business-critical applications.

4 Modernize at your own pace

Outcome: A platform that runs VMs today and opens the door to containers, automation, and AI when you are ready. This is a continuous process, with the pace set by business priority, not the platform.

The starting point: VMs and containers run as native peers on 1 platform. Kubernetes is not bolted onto a separate VM platform or run as a separate silo. There is no forced re-architecture; keep VMs as long as they serve the business.

Three modernization tiers:

- ▶ **Platform modernization:** Already done. Landing your VMs on OpenShift Virtualization puts them on a modern, hybrid-ready platform from the moment migration completes.

- ▶ **Operational modernization:** GitOps, continuous integration/continuous delivery (CI/CD), and unified observability across VMs and containers.

- ▶ **Application modernization:** Application modernization is optional and incremental. Tackle the steps below whenever a specific workload's business case justifies the effort, in any order and at any point, not as a final stage you have to reach:

- Containerize VMs that no longer need a full guest OS.** Lift the workload into a container without rewriting the application: same code, no guest OS overhead, capacity recovered. Use Red Hat's migration toolkit for applications to assess containerization suitability and required changes.
- Refactor for cloud-native environments when the business case warrants.** Restructure into microservices, serverless, or event-driven patterns: these are code changes the platform can support whenever business priorities justify the engineering investment.

Ready to take the next step?

Talk to your Red Hat account team, [explore how Red Hat OpenShift Virtualization can help you recover capacity, reduce cost, and modernize](#) on your terms.

Understand the pressures shaping the decision

Read the companion blog post for the wider hardware, licensing, and skills context behind this checklist, and how a layered efficiency approach helps teams respond on the hardware they already own.

Read the "[Increasing hardware costs? Get more from your VM estate with Red Hat OpenShift Virtualization](#)" blog post.

Go deeper into the mechanisms

Read the [technical companion](#) for the architecture, configuration, and operational tradeoffs behind each efficiency capability, including Free Page Reporting, memory overcommit, scheduler profiles, and hybrid deployment options.



About Red Hat

Red Hat helps customers standardize across environments, develop cloud-native applications, and integrate, automate, secure, and manage complex environments with [award-winning](#) support, training, and consulting services.

f facebook.com/redhat
x x.com/RedHat
in linkedin.com/company/red-hat

North America
1 888 REDHAT1
www.redhat.com

**Europe, Middle East,
and Africa**
00800 7334 2835
europe@redhat.com

Asia Pacific
+65 6490 4200
apac@redhat.com

Latin America
+54 11 4329 7300
info-latam@redhat.com