



10 considerations for Kubernetes deployments

Summary

Container development promises unprecedented portability and scalability in the cloud. In addition, DevOps development and cultural practices offer increased business value and responsiveness. However, there are many questions to consider before starting your first container development project: What operating system should we use? Should we build or buy our Kubernetes platform? How will this new direction affect the organization?

The following 10 considerations can help ensure that your Kubernetes deployment will support your enterprise now—and into the future.

1 Containers are Linux.

You may hear that the operating system doesn't matter when you're running containers. However, the truth is that Linux® is foundational to containers. Containers were first created in Linux, and the technology exists because of key Linux subsystems. When you deploy apps into containers, those apps are running on Linux.

2 Containers are the foundation for Kubernetes.

Kubernetes was also built from Linux, and it uses key Linux constructs, system calls, libraries, and features to manage the infrastructure for and orchestrate containers. When choosing an operating system for your Kubernetes environment, you need a leading, trusted Linux distribution.

3 You do not have to reinvent what already exists.

Kubernetes is more than a mechanism to run your containerized applications. It consists of an application programming interface (API) server, controller, scheduler, and a host of development APIs. Commercial distributions of Kubernetes keep your team from spending time developing what already exists so you can focus on what matters—building portability and scalability into your critical workloads.

4 DIY implementations come with additional ownership.

When you implement a do-it-yourself (DIY) Kubernetes solution, you are responsible for the upgrade and maintenance of new implementations. Your operations team must own a continuous upgrade-and-test cycle that might include alpha or beta features that unexpectedly break compatibility.

5 Not all clouds are the same—but portability is possible.

Different Kubernetes implementations can produce inconsistencies in developer experience, operational models, platform versions, storage, monitoring, and more. However, a consistent Kubernetes implementation ensures that the overall experience is the same, no matter which cloud environment your containers and Kubernetes are running in.

6 Kubernetes is only part of the platform.

Kubernetes only provides the API, orchestration and scheduling, and resource management. To get to a complete application platform, you also need a Linux operating system; a container registry; container networking; container storage, logging, monitoring; and a method to integrate your continuous integration/continuous delivery (CI/CD) pipelines. For richer developer experiences, you will also need advanced capabilities like service mesh, API gateways, application integration workflows, and an integrated developer interface and built-in source control system.

7 Your decisions affect both development and operations.

It's often assumed that a Kubernetes decision primarily affects the development teams. However, Kubernetes introduces entirely new operational concepts and structures that impact day-to-day operations across the enterprise. Before implementing a Kubernetes decision, you should involve your infrastructure, operations, application, enterprise architecture, and lines of business groups.

8 Make Kubernetes easy for your development team to use.

Developers value their time. Deploying to Kubernetes needs to be easy to adopt and use. Be sure to include developer tools that are familiar and support teamwork, and make developer buy-in part of your Kubernetes deployment plan.

9 Kubernetes is still evolving.

Kubernetes is currently in its third evolution, and the platform is only four years old. Since its release, Kubernetes has moved from a high-level concept to a problem-solving, highly scalable, automated solution for both operations and applications. If you are moving to Kubernetes now, it is essential to choose a supported, tested, certified solution that will allow you to evolve without introducing complexity or uncertainty.

10 The right partner can help you accelerate your container adoption.

Expert Kubernetes engineering partners are typically also part of the Kubernetes development community. They can help simplify your container adoption and make sure you are making sound strategic decisions. With a knowledgeable partner, you will implement a solution that will let you deploy your applications across all infrastructure footprints, bare metal, virtualization, private cloud, and public cloud. Industry-leading partners also offer comprehensive portfolios of container products and services, including developer tools, security, application services, storage, and management solutions.

Get started with Red Hat

Learn more about Red Hat's flexible hybrid cloud solutions and find additional resources.

[Explore Red Hat ▶](#)



About Red Hat

Red Hat is the world's leading provider of enterprise open source software solutions, using a community-powered approach to deliver reliable and high-performing Linux, hybrid cloud, container, and Kubernetes technologies. Red Hat helps customers integrate new and existing IT applications, develop cloud-native applications, standardize on our industry-leading operating system, and automate, secure, and manage complex environments. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500. As a strategic partner to cloud providers, system integrators, application vendors, customers, and open source communities, Red Hat can help organizations prepare for the digital future.



facebook.com/redhatinc
@RedHat

linkedin.com/company/red-hat

NORTH AMERICA
1 888 REDHAT1
www.redhat.com

**EUROPE, MIDDLE EAST,
AND AFRICA**
00800 7334 2835
europa@redhat.com

ASIA PACIFIC
+65 6490 4200
apac@redhat.com

LATIN AMERICA
+54 11 4329 7300
info-latam@redhat.com



Copyright © 2019 Red Hat, Inc. Red Hat, and the Red Hat logo are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.