Accelerate your path to cloud-native applications

8 considerations to guide your journey

Contents

Application transformation is a journey

Modern business depends on applications. Across industries, organizations use applications to create differentiated services, interact with customers, and connect partners and employees.

Even so, traditional business models can't keep up with the pace of change brought on by cloud-native technologies and modern application development approaches. These technologies and methodologies help organizations speed innovation and improve agility, giving them a competitive advantage.

As a result, it's not a question of *whether* to digitally transform and modernize, but a question of *when*. For most organizations, creating innovative, differentiated digital services and customer experiences means shifting to a more responsive, agile IT culture. Fast-changing customer and market demands necessitate rapid, flexible application development and delivery models.

However, most organizations still need to maintain at least some of their current infrastructure and processes to make the most of their existing investments. They cannot simply start over and completely rebuild their technology foundation and organizational practices. Cloud-native transformation takes time, and most organizations approach it as an ongoing, iterative journey that incorporates shifts in technology, processes, and culture.

Start with the right technology foundation

Container technologies and DevSecOps approaches are key components in successful cloud-native journeys. Deploying a Kubernetes-powered application platform can help you make the most of these components across hybrid and multicloud environments. The right platform will provide the agility, consistency, efficiency, and scalability needed to build, deploy, run, and manage applications across datacenter, edge, and public cloud infrastructures—all without locking you into a specific public cloud or vendor.

Look for an application platform that provides:

- A trusted and consistent foundation for application deployment across environments.
- A comprehensive set of cloud-native development and operations services and tools.

- Consistent, streamlined security and management capabilities.
- Multiple consumption models, including self-managed on-site, self-managed cloud, and fully managed cloud service options.

While choosing the right technology foundation is a starting point, you also need to consider your people and processes. People are at the core of any enterprise-wide initiative, and cloud-native operations are no different. In order to adopt cloud-native approaches across your organization, all teams—including development, security, and operations—must be on board, participate, and trust each other. Processes move projects from start to finish. Clear processes for creating, deploying, managing, and adapting applications and infrastructure—and incorporating security throughout their life cycles—are essential for broad adoption of cloud-native approaches.

This e-book discusses 8 considerations for adopting a cloud-native approach to application development and delivery.

Consideration 1:

Adapt your culture and practices for the cloud

Adopting cloud-native operations requires your development, business, and IT operations teams to adapt in different ways to streamline and speed application development, delivery, and deployment. Regardless of industry or size, every organization should consider how they will support the collaboration and coordination of activities, technologies, teams, and processes needed for successful cloud-native operations.

While traditional organizational approaches often let teams make independent decisions and move quickly, they create isolated data stores, cloud environments, applications, and processes. This isolation can make it difficult to collaborate, share information and resources, and, ultimately, innovate effectively.

Because successful cloud-native innovation relies on collaboration and rapid hand-offs between team members and teams, an enterprise-wide cloud strategy that brings people, processes, and technology together is essential. A collaborative cloud culture is about more than just new tools and technologies—it's about people. In order to adopt cloud-native practices across your organiation, all teams—including line of business, network, security, operations, development, and infrastructure—must be on board and ready to learn new concepts and skills.

Two key areas to consider are security and management. Moving to cloud-native operations introduces new challenges around security and manageability at scale. Adopting DevSecOps approaches and practices can help you build security into your applications, processes, and platform to better protect your business as you move to the cloud.

DevSecOps extends the collaborative culture of DevOps to incorporate security throughout your application life cycles. It encompasses people, processes, and technology to make security more pervasive in distributed environments. Through DevSecOps, security becomes a shared and enforced responsibility across teams, rather than a set of tasks owned by one team and applied at the end of the development and deployment process. Security, development, and operations teams work together, sharing information, feedback, lessons learned, and insights. This approach allows security to be integrated from the start of application development and infrastructure deployment, increasing protection and reducing risks.

Learn more about adopting DevSecOps

Read **Build a software factory to support DevSecOps** to learn about considerations and guidance for building a successful DevSecOps practice within your organization. →

Consideration 2:

Create a migration plan for your legacy applications

Your cloud-native journey should not focus solely on new application development—it should also incorporate legacy applications. Many traditional applications are critical to business operations and revenue generation. It's important to consider how you will move these applications to the cloud while supporting your business needs now and in the future.

Legacy application modernization often accompanies cloud adoption, allowing both developers and applications to take advantage of cloud features and innovation. By modernizing your existing applications and moving them to the cloud, you can boost scalability, reliability, and security while reducing costs and improving customer experiences.

There are many ways to modernize and move traditional applications to the cloud. Most typically follow 1 of 6 main paths, commonly known at the 6Rs:

- **Retire.** Decommission applications that are no longer needed.
- **Retain.** Leave critical applications as-is until refactoring is required.
- **Rehost.** Lift and shift applications to a cloud with minimal changes.
- **Replatform.** Containerize applications or upgrade operating systems, databases, and other components as needed to allow applications to run in a cloud, without changing the core application code.
- **Refactor.** Rearchitect applications to be cloud-native by moving them to a microservices or serverless architecture, for example.
- **Repurchase.** Move from perpetual application licenses to a Software-as-a-Service (SaaS) model

Most application migration journeys involve rehosting, replatforming, or refactoring applications. While each path has distinct technology, process, culture, and time requirements and delivers different benefits, all will help your organization operate more effectively in a digital world.

All of your applications need not follow the same modernization path. You can choose the path that best fits the characteristics of each application, as well as your organization's current—and expected—needs. You can also opt to make minimal changes to an application now and further modernize it as your requirements evolve.

Plan your application migrations

Read **A phased approach for moving your applications to the cloud** to learn about planning your application modernization and migration journey. →

Consideration 3:

Use application services to speed development

Reusable application components are key to speeding software development, regardless of whether you're building monolithic or cloud-native applications. However, for cloud-native approaches, these components must be optimized for and integrated into the underlying cloud infrastructure to make the most of cloud features and benefits like scalability, reliability, and security.

Ready-made application services that work with your cloud-native application platform let you customize and reuse existing components across your organization. These services can help you build, deploy, and operate applications with security in mind and at scale across hybrid cloud environments.

While DevSecOps practices and container technologies can help speed application delivery and deployment, prebuilt application services can help you further accelerate application development and time to market. For example, developers can use application services specifically built to perform well in a container-based infrastructure. These services take advantage of application platform capabilities like continuous integration/continuous delivery (CI/CD) pipelines, rolling and blue-green deployment approaches, automated scalability, fault tolerance, and more.

Common application services

There are a huge number of different application services available.
Here are some of the most common:

- Application and application programming interface (API) connectivity
- Data transformation
- Service composition and orchestration
- Real-time messaging and data streaming
- Authentication and authorization
- Application frameworks

Gain benefits with a unified platform

A unified development and delivery platform brings together a containerized, cloud-native application platform and ready-made application services and developer tools. Organizations that use a unified platform experience:

190%
more new features delivered per year.[1]

38%
faster development life cycles.[1]

$66.7M
higher revenue per year.[1]

Consideration 4:

Choose the right tool for each task

In cloud-native applications, the development languages or frameworks used are increasingly tailored to specific application needs, resulting in greater complexity and application diversity. There are many languages, frameworks, and tools available, and choosing an appropriate set for each application and task can help streamline development, testing, deployment, and management.

It's essential to select an application platform that supports the right mix of frameworks, languages, and architectures for your organization's cloud-native development goals. For example, you may plan to use the 12-factor approach, domain-based or test-based design, a monolith-first or fast monolith strategy, or a microservices architecture. Your application platform should natively support whichever approaches you choose with a curated set of built-in tools and components. And these tools and components should be continuously updated in line with the latest technological advances.

A platform that includes a comprehensive set of developer tools also lets your development teams choose their preferred languages, frameworks, and toolkits, boosting staff efficiency and satisfaction. In fact, organizations that deploy a unified application development and delivery platform—with built-in developer tools, languages, and frameworks—experience 38% higher developer productivity.[1]

[1] *IDC White Paper, sponsored by Red Hat. "*[The Business Value of a Unified Application Development and Delivery Platform with Red Hat](#)*," December 2022. Document #US49844722.*

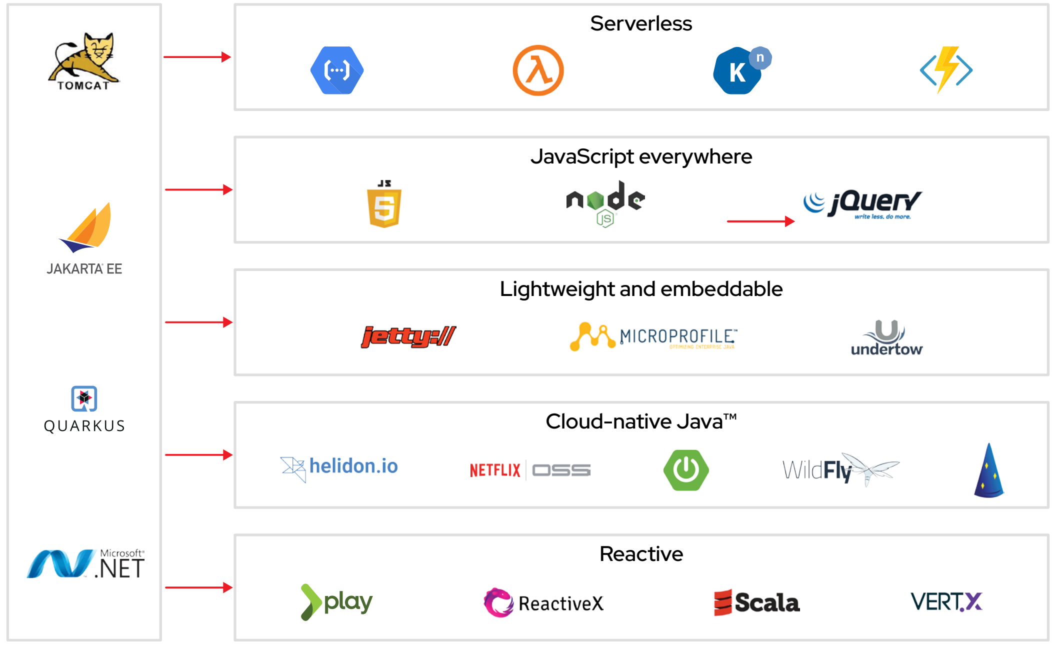


Figure 1. Common developer tools

Consideration 5:

Provide self-service, on-demand infrastructure

Agile development and DevSecOps methodologies help developers rapidly create and update software—once staff have access to the right infrastructure and resources. However, many organizations lack an efficient way of providing infrastructure when and where it is needed. Staff may wait for weeks for the resources to develop, test, or deploy applications to production, slowing overall speed to market for new services and features. This approach is counterproductive to cloud-native operations, especially when infrastructure is generally inexpensive and engineering talent is not.

Self-service and on-demand infrastructure provisioning can give staff the infrastructure they need to do their work while avoiding the risks associated with shadow IT and unauthorized cloud resource use. Even so, for self-service models to be effective, IT operations teams need to have control over and visibility into all aspects of their often dynamic and complex IT environments.

Containers and container orchestration technologies abstract and simplify access to the underlying infrastructure. They can also provide application life cycle management across diverse hybrid environments that encompass datacenters, private clouds, public clouds, and edge deployments.

Containers additionally support application portability. For example, a containerized, cloud-native application can be deployed and run consistently on any cloud provider. This portability lets teams select the best cloud provider for a particular application at any point in time and migrate the application to a different provider as needs and conditions change.

An application platform that includes comprehensive self-service, automation, and application life cycle management capabilities can help you support your development, security, and operations teams effectively and in line with cloud-native principles. Teams can spin up consistent environments on demand, based on preapproved templates. As a result, each team can focus on their main tasks and functions without delays.

Consideration 6:

Apply automation across infrastructure and processes

Speed and accuracy are critical for successful cloud-native operations. Your organization must develop, deliver, and manage security-focused applications and IT infrastructure faster than ever to remain competitive. By streamlining service delivery processes and building the platforms and infrastructure needed for security-focused application development, testing, and deployment, IT operations teams can greatly influence the speed of application delivery.

IT automation can help you connect traditional and cloud-native environments while delivering the operational speed and accuracy you need. It can encompass everything from resource provisioning and retirement to complete life cycle workflows that incorporate management, release engineering, and network and security operations. Using IT automation, you can document, assess, and codify tasks so that they can be combined reliably and repeatedly into workflows to achieve predictable business outcomes. IT automation also helps you create a consistent operational framework across all IT and cloud domains.

IT automation platforms give you a unified foundation for adopting organization-wide automation to support increased IT agility, resilience, and speed. These platforms can automate IT and cloud infrastructure, networks, applications, IT services, and more. They can also automate specific technologies like containers, methodologies like DevSecOps, and broader areas like security, clouds, and development.

Start automating

Organization-wide automation does not happen instantly, and automation is not an all-or-nothing proposition. Here are some tips for getting started:

- Identify your business objectives and connect your automation efforts with business challenges and goals.
- Use incentives to promote cross-team collaboration and coordination across your organization.
- Build a centralized repository for trusted automation content from each team.
- Create a core team of stakeholders—often called a community of practice (CoP) or center of excellence (CoE)—that share automation best practices, experiences, and accomplishments across your organization.
- Choose an automation platform that provides a unified foundation for collaboration, tools, and content across your organization.
- **Train your staff** on automation concepts like source control, testing protocols, and best practices to help them be successful.
- Define what automation success means to your organization and set realistic, measurable goals.
- Take an incremental approach to automation—start small, show value, expand conservatively, and repeat.
- For each successful project, promote the value of automation and share your experience across your organization.

Read **The automated enterprise** to learn more about planning your organization-wide automation strategy. →

Consideration 7:

Implement continuous delivery techniques

Fast-moving digital businesses require short software release cycles to support rapid response to changing conditions. Long release cycles can result in significant delays between identification and resolution of software issues, slow time to market for new features, and less overall business agility. For example, an unresolved bug in a high-traffic application can produce substandard customer experiences, costly outages, and security and compliance issues.

A key tenet of agile development methodologies is *release early and release often*. DevSecOps and continuous delivery approaches bring together developers, operations, quality assurance, and security teams to improve software delivery processes and accelerate software release cycles. CI/CD pipelines provide end-to-end automated delivery systems to support fast feedback loops and rapid, reliable production releases of code changes. These pipelines can incorporate everything from application testing and vulnerability scanning to security and regulatory compliance. The goal is to deliver updates without affecting operational capacity and reduce the risk associated with change.

Continuous integration (CI) is the first step in implementing an automated delivery system. CI systems are build systems that watch source control repositories for changes, run applicable tests on these changes, and automatically build the latest version of an application from the source control change.

Continuous deployment (CD) is the next step. Advanced deployment patterns can help reduce the risk associated with software releases. They can provide an environment for experimentation and innovation, while ensuring control over outcomes and avoiding unintended negative effects on customers and end users.

CI/CD techniques shift software delivery from an off-hours activity with service windows and outages to a routine workday task with no production downtime and continuous application availability. By eliminating downtime for customers and end users, organizations can deliver updates and releases at the speed and frequency needed to meet business and market demands.

Common techniques for zero-downtime software delivery

- **Rolling deployment** is a pattern in which each instance of an application is updated individually, rather than updating all instances simultaneously. This iterative process involves directing traffic away from an individual instance, updating the instance, and then adding the updated instance back to the system load balancer to receive traffic again.

- **Blue-green deployment** is the practice of running two identical environments, one active and the other idle. Changes are rolled out to the idle environment and verified in production. Live traffic is then shifted to the updated environment. You can roll back to the previous version by simply shifting traffic back to the non-updated environment, assuming any needed data transitions are managed accordingly.

- **Canary deployment** also uses 2 identical environments—one active and the other idle—but differs from blue-green deployment in the way roll outs are controlled. Changes are released into the idle environment, and then a small subset of users is redirected to the updated environment to test the release in production. If the release is stable, traffic is incrementally shifted to the updated environment. Outcomes are continuously monitored and verified until all traffic is eventually sent to the updated environment.

Consideration 8:

Move to a more modular architecture

Microservices architectures break applications down into sets of functional capabilities. These functional microservices are separate from each other but work together to accomplish the same tasks as monolithic applications. They are typically lightweight, independently scalable, and shareable across multiple applications. Microservices architectures align well with cloud-native approaches and are often adopted during application modernization initiatives. In fact, container-based application platforms are an optimal foundation for microservices architectures.

Adopting a microservices architecture can provide many benefits, especially for large teams and organizations that release changes to production very frequently. Microservices allow you to break each application down into individually managed and deployed pieces. The life cycle of each microservice can be maintained separately—potentially by different teams—without affecting other microservices or breaking the entire application.

Take a phased approach to microservices architecture adoption. Rebuild your traditional applications incrementally over time to gradually move functionality from your old application architecture to your new one. During the redevelopment process, you can also upgrade underlying technologies and add in new cloud-native services and capabilities like artificial intelligence and machine learning (AI/ML), analytics, autoscaling, serverless functions, and event-driven architecture.

It's important to understand that not every application will benefit from being rearchitected as microservices. Assess each application to ensure that rearchitecting it makes sense and will deliver value to your organization. For example, the cost of rearchitecting applications with a small number of users or limited need to scale may actually be more than the benefits. Your goal should be to choose the architecture and environment that delivers the best return on investment for each application.

Create an efficient service architecture

Read **Service mesh or API management?** to learn more about implementing microservices, APIs, and service architectures. →

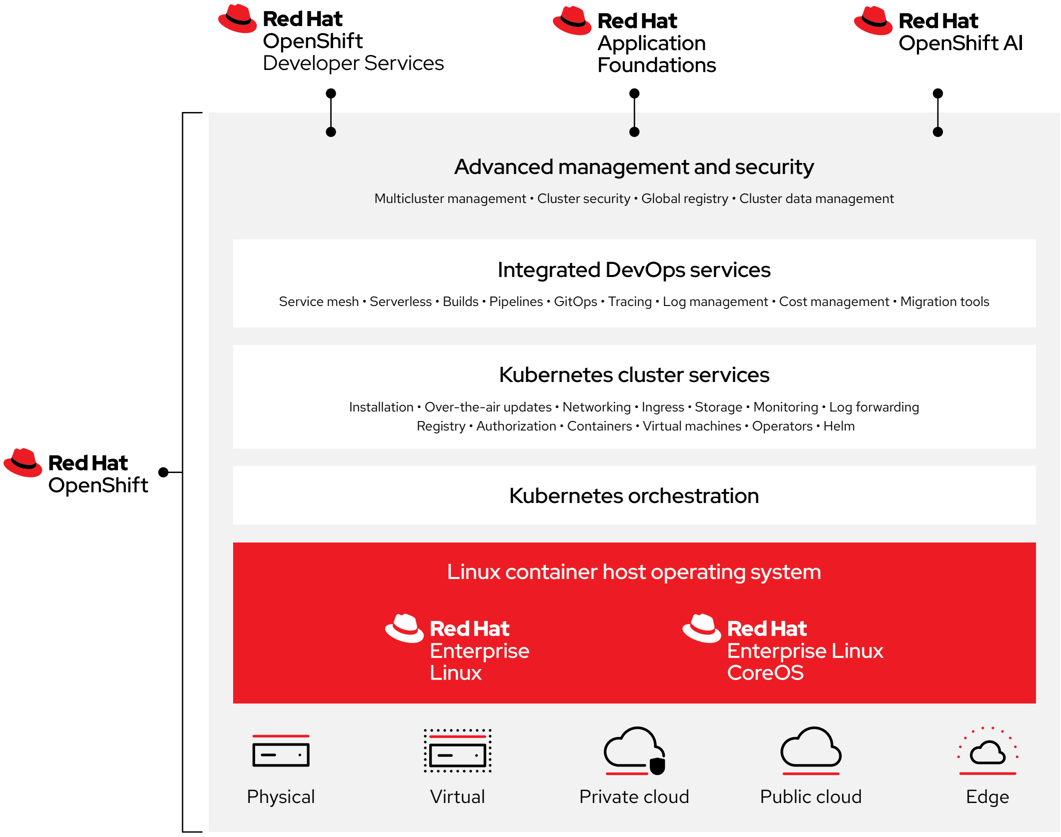
Plan your path to cloud-native applications with Red Hat

No matter where you are on your cloud-native journey, Red Hat can help you get the most benefit from your efforts.

As an expert in open source software, cloud-native technologies, and Kubernetes, Red Hat can help you build an agile, cloud-native IT environment to support evolving business demands. We offer a complete, unified, open hybrid cloud foundation for your cloud-native journey.

- Red Hat® OpenShift® provides a unified, enterprise-ready application platform for cloud-native innovation. It runs consistently across hybrid and multicloud environments, giving you a comprehensive set of tools, services, and capabilities for rapidly building, deploying, and managing existing and cloud-native applications at scale and with security.
- Red Hat Application Foundations delivers a comprehensive set of components for developing and modernizing software. It can be used with applications that run on-site or in the cloud and, when combined with Red Hat OpenShift, it creates a platform that streamlines execution across the entire application life cycle.
- Red Hat Runtimes is a set of products, tools, and components for developing and maintaining cloud-native applications. It offers lightweight runtimes and frameworks like Quarkus for highly-distributed cloud architectures, including microservices.

Our hybrid cloud application platform spans on-site datacenters, private and public clouds, and edge infrastructure to deliver both the consistency and flexibility you need to transform on your terms. The components and platforms in this foundation are offered both as self-managed products and as cloud services that offload management and maintenance operations, allowing you to focus on your business priorities. These cloud services can help you rapidly build a hybrid cloud environment and focus on your business priorities, rather than administering your application platform.



Red Hat OpenShift Developer Services

Red Hat Application Foundations

Red Hat OpenShift AI

Red Hat OpenShift

Advanced management and security

Multicluster management • Cluster security • Global registry • Cluster data management

Integrated DevOps services

Service mesh • Serverless • Builds • Pipelines • GitOps • Tracing • Log management • Cost management • Migration tools

Kubernetes cluster services

Installation • Over-the-air updates • Networking • Ingress • Storage • Monitoring • Log forwarding
Registry • Authorization • Containers • Virtual machines • Operators • Helm

Kubernetes orchestration

Linux container host operating system

Red Hat Enterprise Linux

Red Hat Enterprise Linux CoreOS

Physical

Virtual

Private cloud

Public cloud

Edge

Figure 2. Red Hat’s complete, unified, open hybrid cloud foundation

Ready to start your cloud-native journey?

A cloud-native approach to application development and delivery can help you innovate rapidly and compete effectively in a digital world. Red Hat offers the products, expertise, and services to make your journey successful. Move at your own pace while increasing efficiency, resiliency, and flexibility with a complete, unified foundation for all of your applications.

Discover our cloud-native development solutions today.

Simplify your cloud-native journey

Red Hat Consulting experts can work with you to evaluate and implement cloud-native solutions that help you reduce costs, improve efficiency, and speed development. Our experts can also help you, your team, and your organization develop the practices, tools, and culture needed to rapidly build and adapt cloud-native applications across your organization.

Learn about our consulting services →

Get the skills needed for cloud-native success

Red Hat offers a comprehensive curriculum of training courses to help your organization fill skills gaps and address business challenges. Available in multiple tiers, a Red Hat Learning Subscription gives you unlimited access to Red Hat Training courses, including online self-paced and instructor-led courses, cloud-based labs, and certification testing.

Find your skills path →

Red Hat