8 steps for cloud-native application development success

Taking the cloud-native application development journey

In today’s digital world, applications are the key to interacting with customers, partners, and employees. Application development can make or break any organization’s success. Cloud-native development enables organizations to give their developers the flexibility to build, deploy, and run applications more securely—anywhere they are needed—including across multiple clouds.

Taking the cloud-native journey requires a methodical approach that enables a transition from traditional software development to a cloud environment. The practical course of action is a gradual shift in culture, processes, and technology toward cloud-native development. This approach includes a step-by-step transition to microservices architecture, breaking down your monolithic applications into smaller components (at your own pace), and setting the foundation for cloud-native development success.

Why take a cloud-native application development journey?

The move to cloud-native application development is achieved from a combination of practices, technologies, processes, and mindsets, ultimately delivering several high-value benefits to the development team and the entire organization, including:

- **Faster development** that breaks applications into portable reusable components, speeds up the development life cycle, and reduces time to market.
- **Greater scalability** in a cloud environment that helps you scale applications more easily and cost-effectively.
- **Improved efficiency** using automated processes that makes the information technology (IT) team more efficient and reduces costs.
- **Better product quality** with DevOps and continuous delivery that helps reduce and eliminate software bugs and improve application quality.
- **Enhanced innovation** that gives you easier access to on-demand infrastructure and developers more freedom to focus on innovation.
- **Cloud-agnostic cost-efficiency** with microservices and containers that creates portable applications that you can deploy across multiple cloud vendors, helping avoid cloud vendor lock-in.
- **Future-ready applications** with cloud-native development that allow users to update applications more quickly and easily to meet changing market demands.
- **Higher return on investment (ROI)** by using cloud-native development that allows you to integrate monolithic applications with modern applications and retain the value of your legacy applications.
8 steps for cloud-native application development success

This outline identifies eight steps that an organization should consider when looking to adopt a cloud-native approach to application development.

Step 1: Expand culture and practices for cloud environments—Establish an enterprise-wide cloud strategy and encourage willingness and trust to embrace a more integrated and collaborative approach to developing and delivering applications in a cloud environment.

Step 2: Speed existing applications using microservices—Move from your existing monolithic architecture to a more modular, microservices-based architecture with application programming interface (API)-based communication.

Step 3: Use application services to speed development—Take advantage of existing, ready-to-use application services that have been optimized and integrated into the underlying container-based infrastructure—available as Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), or integration PaaS (iPaaS) offerings.

Step 4: Choose the right tool for the right task—Use a container-based application platform that supports the right mix of frameworks, languages, and architectures to support cloud-native development.

Step 5: Provide self-service, on-demand infrastructure—Use containers and container orchestration technology to abstract and simplify access to the underlying infrastructure and provide robust application life cycle management across various infrastructure environments.

Step 6: Automate IT to accelerate application delivery—Deploy automation tools to create repeatable processes, rules, and frameworks that can replace or reduce labor-intensive human interaction that delays time to market.

Step 7: Implement continuous delivery (CD) and advanced deployment techniques—Adopt DevOps and CD approaches to unite developers, operations, quality assurance, and security teams to improve software delivery processes.

Step 8: Advance to a more modular architecture—Move away from a traditional, monolithic approach by using a microservices-based architecture with a container-based platform to separate applications into components that perform specific services.

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.

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

Copyright © 2021 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.