Developing apps in containers: 5 topics to discuss with your team

Sometimes talking to your development team can be difficult. You might be asking a lot from your developers with a limited budget, maintenance of existing systems, and tight deadlines. While you empower them to learn new technologies and approaches and give them the tools they need for success, the myriad of new information can be daunting.

Red Hat can help in your journey to containers and containerized application services and middleware, allowing you to be more efficient, innovative, and responsive to business needs. LogistiCare, for example, had systems that could not keep up with its growth. The company came to Red Hat looking for innovative ways to increase the speed of development and deployment while increasing system performance. With Red Hat® OpenShift® Container Platform and Red Hat Middleware, LogistiCare created a new app development environment that resulted in increased release cadence and reduced operational costs.[1]

Here are five topics to bring to your app dev team to gauge how moving to containers and containerized application services and middleware can help increase productivity and deliver solutions to the business faster. When asking the questions for each topic, consider the list of keywords or phrases the developers might use in their answers. If you hear these phrases, it is probably time to start looking into a Red Hat container deployment model.

1 Faster deployment and delivery of apps

Development teams across all industries are being pushed to build new applications and services to enable business innovation and transformation. Here are some questions to guide the conversation with your dev team.

1. How complex is it for you to build an app in containers?

2. How long does it take you to apply a fix, update, or enhancement to an existing app?

3. Are you using continuous integration/continuous delivery (CI/CD) deployment pipelines in your application development process?

4. Do you frequently face issues with deployment in higher environments that do not occur in the development environment?

2 Support for multiple languages and frameworks

Language and framework support can be difficult. Being able to work multiple development languages and frameworks allows you to avoid being weighed down with old legacy systems, and it helps you attract a wider range of top talent. Here are some questions you can ask.

1. How are you exploiting microservices in your existing and new apps?

2. Are you using well-established frameworks and languages?

3. What languages and frameworks do you see as leading edge and designed for microservices?

4. Does your cloud-native application environment support the optimization of existing apps as well as the development of modern apps?

---

1 *Red Hat case study. "LogistiCare accelerates flexibility and speed of business," May 2017. https://www.redhat.com/en/resources/logisticare-case-study.*

Keywords and phrases: Dev time is in months or years, app updates process is complex and long, lack of CI/CD pipelines, need for A/B testing, differences between deployment environments.

Keywords and phrases: Use of a single language or framework for all development work, lack of leading-edge languages and frameworks that support modern development techniques, unable to optimize existing applications to run on containers and Kubernetes, existing solution only supports new development of applications.

3 Ability to work in hybrid cloud settings

Moving to a hybrid cloud setting has many benefits, but mainly it allows your development team to innovate rapidly and safely. Your containers can run in multiple cloud settings. If your dev team is thinking about moving to hybrid cloud, consider these questions.

1. Are you developing apps that take advantage of one or more cloud providers?

2. Are you developing applications that span across many clouds?

3. How much time are you spending on making your app work across clouds?

4. Are you developing apps made up of on-premise and on-cloud functionality?

Keywords and phrases: Unable to develop on the cloud, development is only on-premise, use of a single cloud provider, cloud provider lock-in, unable to develop hybrid or multicloud applications.

4 Application services and middleware optimized for Kubernetes

Kubernetes is the industry leader for container orchestration today.[2] Having your development team work with Kubernetes benefits your business by increasing the team's efficiency. These questions are helpful to consider.

1. Have the application services and middleware you are using in your applications been engineered to work together and optimized for Kubernetes?

2. Do the application services and middleware you are using in your applications have a long track record of production deployments?

3. How are you bringing security to your apps? Are you using integrated security capabilities that work across clouds and on-premise?

4. How hard is it to monitor and troubleshoot the deployed services?

Keywords and phrases: Current application services and middleware being used have not been engineered to work together and optimized for Kubernetes, importance of app security, current app services and middleware do not have a long track record of production deployments on Kubernetes, app security is fragmented, using disparate security mechanisms across applications, application security is done in an ad hoc manner using specific cloud vendor mechanisms, using disparate monitoring and troubleshooting mechanisms across applications.

---

2 *Vaughan-Nichols, Steven. "5G depends on Kubernetes in the cloud." ZDNet, 2 May 2019. https://www.zdnet.com/article/5g-depends-on-kubernetes-in-the-cloud/.*

5 Developer tools and resources for containerized environments

Career growth and education are important to a development team. Red Hat's services allow for developers to ask questions, attend workshops, and share collateral to help make the transition to deploying apps in containers. Here are some questions to ponder.

1. Are you using an integrated development environment (IDE) for the development of cloud-native apps?

2. Is your IDE designed for the development of apps on containers and Kubernetes?

3. How long does it take you to develop, deploy, and deliver an app on Kubernetes using your current development tools?

4. How much time are you spending on self-training on a daily basis?

5. How important is continuing your learning about the latest application trends via formal education, training, and experiential hands-on activities?

6. How do you currently get technical help for your application development needs?

Keywords and phrases: IDE does not support containers and Kubernetes, IDE does not run on containers and Kubernetes, IDE is not browser-based, IDE requires installation on developer's desktop, app development time using current development tools could be improved, technical support is slow or unresponsive, lack of education and training resources, unavailability of experiential hands-on activities for learning new cloud-native app development.
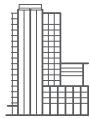
About Red Hat

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
europe@redhat.com

Asia Pacific
+65 6490 4200
apac@redhat.com

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

redhat.com
#F19129_0919