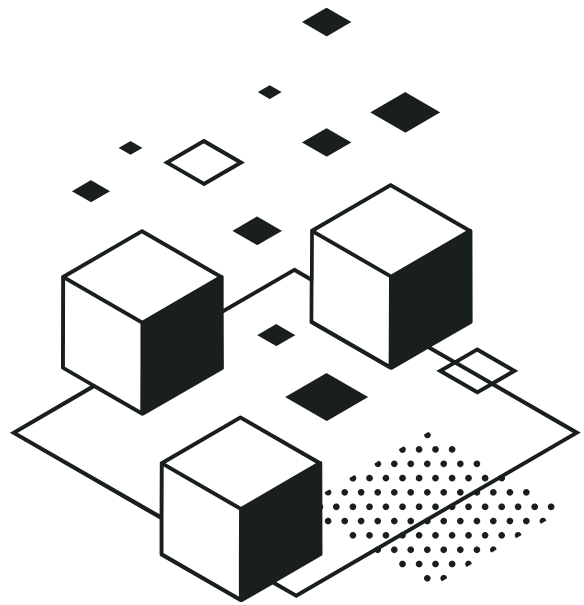


Microservices

Microservice architectures are the 'new normal'. Building small, self-contained, ready to run applications can bring great flexibility and added resilience to your code. Spring Boot's many purpose-built features make it easy to build and run your microservices in production at scale. And don't forget, no microservice architecture is complete without [Spring Cloud](#) – easing administration and boosting your fault-tolerance.



What are microservices? **Why build microservices?**



approach to software whereby application code is delivered in small, manageable pieces, independent of others.

lead to many additional benefits, such as easier maintenance, improved productivity, greater fault tolerance, better business alignment, and more.

Microservices with Spring Boot

With Spring Boot, your microservices can start small and iterate fast. That's why it has become the de facto standard for Java™ microservices. Quickstart your project with Spring Initializr and then package as a JAR. With Spring Boot's embedded server model, you're ready to go in minutes.

TRY OUR QUICKSTART GUIDE

Microservice resilience

The distributed nature of microservices brings challenges. Spring helps you mitigate these. With several ready-to-run cloud patterns, [Spring Cloud](#) can help with service discovery,



Cloud

gateway.

Build streaming data microservices with Spring Cloud Stream

Spring Cloud Stream makes it easy to consume and produce events, no matter which messaging platform you choose. Spring Cloud Stream connects your microservices with real-time messaging in just a few lines of code, to help you build highly scalable, event-driven systems.

[Get started with Spring Cloud Stream](#)



Manage your microservices

Spring Boot's optional instrumentation framework, [Micrometer](#), sends metrics straight to Prometheus, Atlas, and more to provide valuable insights. This is complemented by [Spring Cloud's](#) Sleuth and Zipkin projects which offer distributed tracing so that you can follow along with what's happening in real-time.

[Get started with Micrometer on Spring Boot](#)

Microservices on Cloud Foundry

The small, stateless nature of microservices makes them ideal for horizontal scaling. Platforms like TAS and PKS can provide scalable infrastructure to match, with and greatly reduce your administrative overhead. Using cloud connectors, you can also consume multiple backend services with ease.

CLOUD  FOUNDRY



Ready to get started?

TRY THIS TUTORIAL

More resources

Migrating to Cloud-Native Application Architectures

Matt Stine

Thinking Architecturally

Nate Schutta

Cloud-Native Java: Designing Resilient Systems with Spring Boot, Spring Cloud, and Cloud Foundry

Josh Long & Kenny Bastani



Get ahead

VMware offers training and certification to turbo-charge your progress.

[Learn more](#)

Get support

Tanzu Spring Runtime offers support and binaries for OpenJDK™, Spring, and Apache Tomcat® in one simple subscription.

[Learn more](#)

Upcoming events

Check out all the upcoming events in the Spring community.

[View all](#)

Why Spring

Microservices
Reactive
Event Driven
Cloud
Web
Applications
Serverless
Batch

Learn

Quickstart
Guides
Blog

Community

Events
Team

Solutions

Tanzu Spring Runtime
Spring Consulting
Spring Academy For Teams
Spring Advisories

Projects

Training

Thank You

Get the Spring newsletter



Copyright © 2005 - 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.



Apache®, Apache Tomcat®, Apache Kafka®, Apache Cassandra™, and Apache Geode™ are trademarks or registered trademarks of the Apache Software Foundation in the United States and/or other countries. Java™, Java™ SE, Java™ EE, and OpenJDK™ are trademarks of Oracle and/or its affiliates. Kubernetes® is a registered trademark of the Linux Foundation in the United States and other countries. Linux® is the registered trademark of Linus Torvalds in the United States and other countries. Windows® and Microsoft® Azure are registered trademarks of Microsoft Corporation. “AWS” and “Amazon Web Services” are trademarks or registered trademarks of Amazon.com Inc. or its affiliates. All other trademarks and copyrights are property of their respective owners and are only mentioned for informative purposes. Other names may be trademarks of their respective owners.