2023-04-26-be-mountain

Week 15 Research

Cody Schliebe

Spring Boot vs Spring

Spring is an open-sourced framework that provides all-inclusive infrastructure support to building Java applications. It's lightweight, loose-coupled, and includes powerful features such as dependency injection and aspect-oriented programming. It also comes with a number of modules built for different tasks, such as Spring Test, Spring JDBC, and Spring ORM. Earlier java developments required tons of boilerplate code to do something as simple as inserting a record into a data source, while with JDBC the same thing can be accomplished with only a few lines of code.

Spring Boot was built on top of spring, and allowed elimination of further boilerplate configurations that were necessary in the set-up of an application, using autoconfiguration. It also performs rigorous metrics and health checks, taking care of external configurations so you don't have to. It includes embedded servers to reduce deployment complications.

Spring's applications are loosely coupled, meaning it is great for building enterprise applications. Spring Boot's applications are standalone, meaning Spring Boot is much more suited to building REST APIs.[1]

Gradle vs Maven

Both Gradle and Maven are automation systems that handle dynamic dependencies using third-party dependency caches and POM metadata. Gradle uses a domain-specific language based on the language Groovy to build its project configuration, while Maven uses XML. Gradle also determines the order of tasks run based on a graph. Maven is used mostly for Java, but can be used for other languages such as Ruby, C#, and Scala. Gradle currently works with Java, Scala, and Groovy, and promises support for other languages in the future. It is the primary dependency system used in the development of Android applications.

While both similarly handle dependency libraries, Maven loads them linearly based on the XML file, which can be slow when the file becomes larger. Gradle uses incremental builds based on the graph, skipping any tasks that aren't updated, meaning faster load times. Gradle is also more suited for customization on larger projects, while Maven's XML file can tend to get messy and hard-to-read pretty quickly.

Both systems are great in their respective places. Gradle is more powerful and does better with large, complex projects, while Maven is easy-to-use and perfect for smaller projects.[2]

1 https://www.upgrad.com/blog/spring-vs-spring-boot/

2 https://stackify.com/gradle-vs-maven/