Java magazine   Book Reviews

Enterprise Java Microservices

Reviewed by Andrew Binstock

February 27, 2020

It is unclear whether the current interest in microservices architecture is a fad spurred on by the availability of easily accessible cloud services or whether it will endure and become a standard model of application design. Whatever its ultimate fate, it's clear for the time being that many Java developers will be called on to create and implement microservices. If you are one of them and have no real idea of where to start, this slim volume should give you what you need to get up and running, that is, if you are an experienced developer. The book states up front that it is intended for developers with at least four years of experience in writing Java.

Finnigan starts with basic definitions of microservices, a discussion of the various technologies, and the concepts and designs that are foundational to microservices. Topics in this first half of the book include building REST endpoints and choosing a small-scale application server, or what the author refers to as JeAS (Just enough Application Server). The term is his creation and after examining several possible servers for this role, he ultimately chooses Red Hat's Thorntail. While I can't argue with the technology choice per se, Thorntail is the project for which the author is responsible at Red Hat. While the book stands up fine with this server, there remains a thread of doubt throughout—would this be done differently with another app server? Am I learning microservices or Thorntail-specific microservices? The correct answer is some of both. Most of the explanations are indeed portable to other platforms.

For example, a lot of the early discussion, such as the differences in design and architecture between monoliths and microservices, are well explained and are valid independent of the implementing technology. One of the topics Finnigan pays close attention to both in the design and implementation phases is testing, which is often overlooked in discussions of microservices—replaced too often with hand waving and comments bemoaning the difficulty.

In the second half of this book, Finnigan presents the implementation in considerable detail. The explanations are clear and easy to follow, in good part because he has skillfully broken the topic into easily approachable tasks.

In these explanations, Finnigan favors widely used libraries and packages, so this is not strictly a Red Hat–only tutorial. For example, Finnigan discusses using Netflix Hystrix, a library described in the September 2018 issue of this magazine. And he includes a section on data streaming using Kafka (inexplicably missing from the book's index). Later, he explores tools that help deploy and manage microservices, including security tools. He even explores, albeit briefly, handling payments with Stripe.

Enterprise Java Microservices

By Ken Finnigan
272 pages

When it comes to deploying the sample app, Finnigan shows multiple possibilities beyond just the microservices option. For example, he sagely explains deploying as a monolith and as a smaller monolith with some parts implemented as services. And he covers deployments on hybrid clouds. This thoroughness runs throughout the book and makes it likely that by the time you've read through the more than 200 pages, you'll have a good command of microservices, their architectural requirements, implementation details, and deployment challenges. That's a lot of value.

My only concern is that Red Hat is a little too prominent in the discussions. Deployment is foremost on the OpenShift cloud, which, with all due respect, seems an unlikely target for most readers. However, the author is good enough to frame the discussions so that you can easily map what he's saying to non-Red Hat tools and platforms.

This book is ideal for Java developers who are being asked to move from monoliths to microservices. Their natural temptation might be to study some corner of the migration most applicable to them and miss the fact that microservices require an entirely new perspective on the design and implementation of enterprise software. This quick read will make those programmers conversant with the issues and familiar with some of the leading solutions: a solid recommendation for a well-written book.