

Crowdcast starts soon!

Quarkus: The blue hedgehog of Java web frameworks

Dr. Marco Bungart Software Engineer

2020-11-18



IBM & Consol present: Openshift + Quarkus = ♥ Quarkus: The blue hedgehog of Java web frameworks

Dr. Marco Bungart Software Engineer

2020-11-18

Your Host

Dr. Marco Bungart

- 2008 2013: Studied Bioinformatics/Computer Science in Jena
- 2013 2018: Ph.D. student in Kassel
- Since 2018: Software Engineer at ConSol
- Twitter, github, bitbucket, stackoverflow, ...: turing85
- Interests: Keycloak, GraalVM, Quarkus





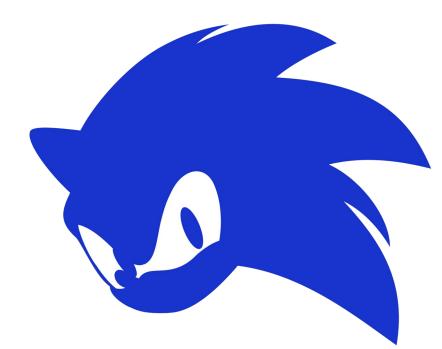
Quarkus!

Questions

- Who of you is working with JEE/Jakarta?
- Who of you is working with Spring-Boot?
- Who of you have heared of:
 - •GraalVM?
 - •Quarkus?
- Who of you have used Quarkus in
 - •A private project?
 - •In production?

What is Quarkus?

- "A Kubernetes Native Java stack tailored for OpenJDK HotSpot and GraalVM, crafted from the best of breed Java libraries and standards." quarkus.io
- Lightweight in execution (small memory footprint)
- Compilation to native binary possible through GraalVM
 - Even lower memory footprint
 - •Single instance may have worse performance, but more deployments possible due to lower resource consumption → net gain
- Up & running within minutes



Source: https://1000logos.net/sonic-logo/

Why Quarkus? (1)

- Developer-joy front & center
 - Developed by developers, for developers
 - Excellent Maven- and gradle-integration
 - Powerful dev-mode, tight feedback loop
- Curated list of dependencies, "best of the breed"
 - Red Hat- and Community-developed
- Apps (re)start within milliseconds

Why Quarkus? (2)

- Based on Java EE/Jakarta & Microprofile
 - •No proprietary dependencies, all based on open standards
 - •Dependency Injection, JPA, OpenAPI, Metrics, Tracing, Fault Tolerance, ...
- Reactive programming supported (through Mutiny)
 - Internally based on Vert.x
- •Not standardized, but compatible with well-known reactive frameworks, like <u>RxJava</u> & <u>Project Reactor</u>

Let me go fast

Bootstrapping a Quarkus Application

```
Maven:
mvn io.quarkus:quarkus-maven-plugin:1.9.2.Final:create \
    -DprojectGroupId=my-groupId \
    -DprojectArtifactId=my-artifactId \
    -DprojectVersion=my-version \
    -DclassName="org.my.group.MyResource"
Gradle:
mvn io.quarkus:quarkus-maven-plugin:1.9.2.Final:create \
    -DprojectGroupId=my-groupId \
    -DprojectArtifactId=my-artifactId \
    -DprojectVersion=my-version \
    -DclassName="org.my.group.MyResource" \
    -Dextensions="resteasy-jsonb" \
    -DbuildTool=gradle
```

What did we just do?

Let's take a look!

Adding dependencies

List extensions:

- ./mvnw quarkus:list-extensions
- ./gradlew listExtensions

Add extension:

- ./mvnw quarkus:add-extension -Dextensions="hibernate-validator"
- ./gradlew addExtension -extensions="hibernate-validator"
- More informations on extensions & guides: <u>quarkus.io/guides</u>

Agenda

What will we talk about today?

- Bootstrapping a project (already done)
- •Implement two microservices for users and colors:
 - •Each microservice has their own database
 - Each user will have a favorite color in the end
- While doing so, we take a look at:
 - •Basics: general tooling, Rest, DI, Persistence, Exception Mapping, Validation, OpenAPI, Metrics
 - REST-Communication between services
 - Tracing with Jaeger
 - Health Checks & Resiliency

What will we not talk about today?

- Native compilation
- Reactive programming with Mutiny
- Messaging Queues
- Camel-Integration
- Quarkus for AWS Lambda
- Authentication/Authorization (next talk on 2020-12-02 will cover this part!)
- •Advanced topics, e.g. context propagation, multi-module + dependency injection, ...

Live Demo



Source: https://makeameme.org/meme/lets-pray-to-9000c9f697

consol.com

Resources

Resources

- quarkus.io
- Quarkus zulip chat
- Code shown in this talk (github)

consol.com

I want more

We (Consol) offer quarkus workshops. If you are interested, please visit:

https://www.consol.de/software-engineering/quarkus/

Recap

Recap

- •Quarkus: why, what & how?
- Bootstrapping a new application & add new dependencies
- Development of two REST-microservices: users & colors:
 - •Basics: general tooling, Rest, DI, Persistence, Exception Mapping, Validation, OpenAPI, Metrics
 - •REST-Communication between services
 - Tracing with Jaeger
 - Health Checks & Resiliency

Next talk!

Wed, Dec. 2nd 2020, 4 PM CET (in 2 weeks)

Application Security with Keycloak and Quarkus: Please do not STATE your business

Questions?

Thank you!



ConSol
Consulting & Solutions Software
GmbH

St.-Cajetan-Straße 43 D-81669 Munich Germany Tel.: +49-89-45841-100 info@consol.de www.consol.com

Twitter: @consol de