Red Hat
Developer

Quarkus & Kubernetes
The fellowship of Cloud-Native Java

Ana-Maria Mihalceanu
Developer Advocate

Red Hat

Hello! I am Ana

ammbra1508

Java Champion, Certified Architect

Developer Advocate @ Red Hat

Passionate about solving complex scenarios involving Java-based frameworks and multiple cloud providers.

Why create Cloud-Native applications?

Building Architectures Open to Concurrent Changes

Quicker iteration of features or fixes.

Design loosely coupled services

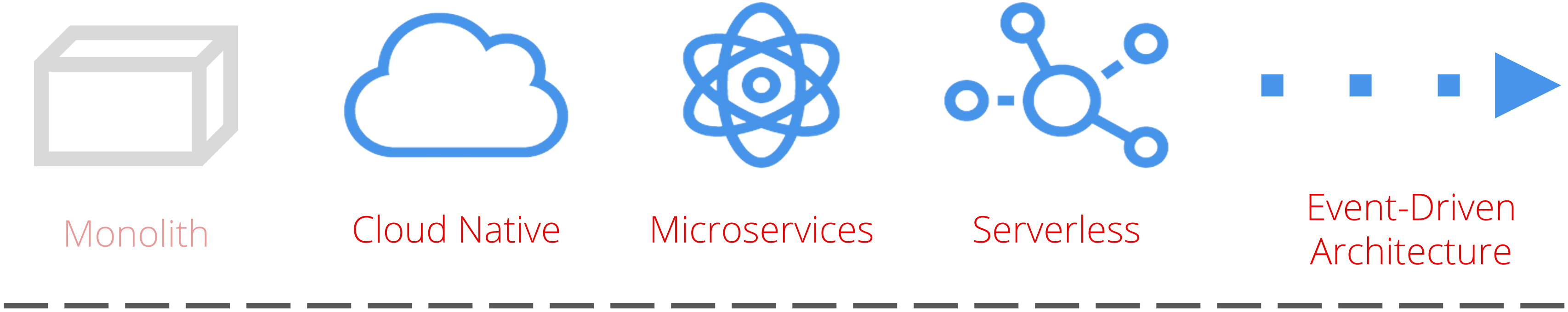
Work with delightful programming models

Develop using languages and frameworks with a fast learning curve.

Cost-Efficient Autoscaling

Package applications as lightweight containers.

Kubernetes-Native Java



Monolith

Cloud Native

Microservices

Serverless

Event-Driven Architecture

kubernetes

Istio

Knative

The Challenge

> As our platform evolved and the complexity increased, our developer process was becoming more of a chore and dependencies between technology teams, front-end, back-end, and operations, started to become a bottleneck.

> After some analysis, we quickly uncovered three major pain-points:
>
> - It’s too hard to setup the environment
> - The learning curve was too steep
> - Our CI/CD process was automated but not enough

Source:
https://quarkus.io/blog/logicdrop-customer-story/#the-challenge

How to adopt Quarkus

“”

> Making objective choices (for the company and for developers who have a passion for the framework), on a new framework for our company was very difficult! We cannot rely solely on the advice of the Quarkus expert on our team to make the choice. This is the reason why we started with a POC, with the objective that if we would face some impossible situation, we would go back to Spring Boot without hesitation!

Source:
https://quarkus.io/blog/decathlon-user-story

From challenges to successful changes

Change is Inevitable

“”

- DI became predictable not magical (big win)

- Mongo and Panache did just what we needed nothing more

- Kubernetes configurations could be kept closer to the service

“”

- Mutiny made writing reactive services way easier for everyone

- Security worked out-of-the-box and was well defined (big win)

- REST clients helped isolate services cleaner and better

Source:
https://quarkus.io/blog/logicdrop-customer-story/#change-is-inevitable

Change is Inevitable

“”

- OpenAPI and Swagger worked without tweaking

- Intimate knowledge of Maven was no longer required

- Testing became easier and consistent

“”

- CI/CD orchestrates all services together rather than each one separately

- No more bloated dependencies

- Livecoding just made everything better

And the list keeps going…

Source:
https://quarkus.io/blog/logicdrop-customer-story/#change-is-inevitable

Quarkus - A proven technology

WHAT IS QUARKUS?

QUARK: elementary particle / **US**: hardest thing in computer science

Differentiators

Container First

Tailors your app for HotSpot & GraalVM

Fast boot time and low RSS memory

Serverless fit

Developer Joy

Live coding

Unified configuration

Continuous Testing & Dev Services

Unifies Imperative & Reactive

Combines blocking and non-blocking

Built-in event bus

Best of Breed Libraries & Standards

90+ extensions

“Powered by Quarkus” applications

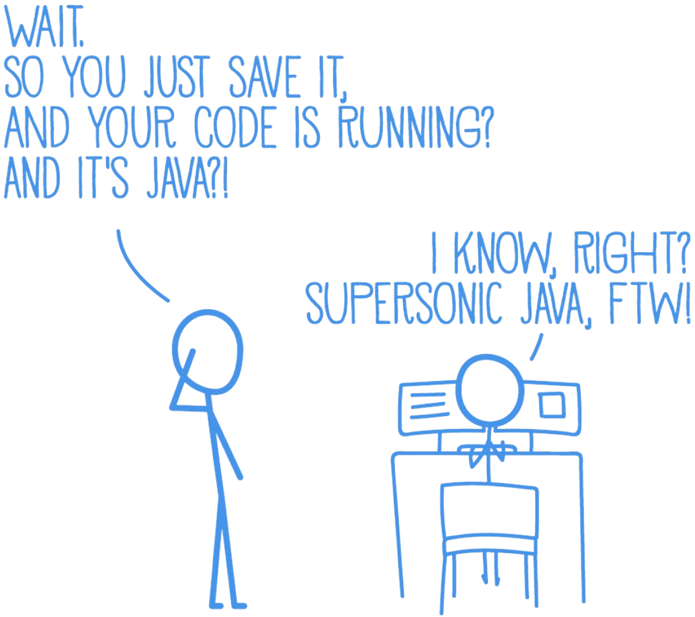
How those differentiators impact your work?

Gratifying Development Process

Developer Joy

A cohesive platform for optimized developer joy:

- Based on standards that **make development and learning codebases easier**
- Unified configuration and Live coding
- **Continuous Testing** enables consistent testing.
- **Dev Services** offer zero config setup for some databases and other services, like Kafka.
- Quicker iteration over features and fixes
- No hassle native executable generation



Delightful Programming Models

Unifies Imperative and Reactive

```
@Inject
SayService say;

@GET
@Produces(MediaType.TEXT_PLAIN)
public String hello() {
    return say.hello();
}
```

```
@Inject @Stream(”kafka”)
Publisher<String> reactiveSay;

@GET
@Produces(MediaType.SERVER_SENT_EVENTS)
public Publisher<String> stream() {
     return reactiveSay;
}
```

- Combine both Reactive and imperative development in the same application
- Inject the EventBus or the Vertx context
- Use the technology that fits your use-case
- Key for reactive systems based on event driven apps

Integrations Made Easy via Extensions

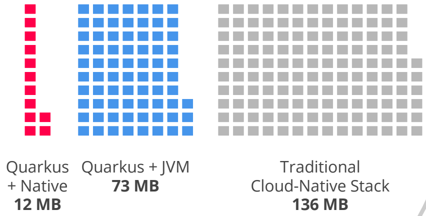
Quarkus Universe: Best of Breed Frameworks & Standards

- Eclipse Vert.x
- Hibernate
- RESTEasy
- Apache Camel
- Eclipse MicroProfile
- Netty
- Kubernetes
- OpenShift
- Jaeger
- Prometheus
- Apache Kafka
- Infinispan
- Flyway
- Neo4j
- MongoDB
- MQTT
- KeyCloak
- Apache Tika

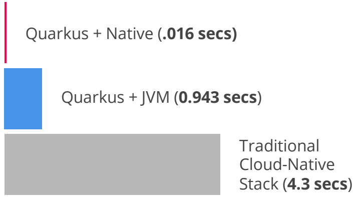
Dynamically Scale Up and Down Quickly

Container First

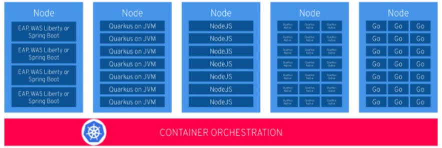
Reduced Memory Footprint



Fast Startup Time



Smaller Disk Footprint



*Memory (RSS) in Megabytes, tested on a single-core machine

** Time for first response for a REST based application

***https://developers.redhat.com/blog/2017/03/14/java-inside-docker/

****An article referencing fast startup: https://quarkus.io/blog/vodafone-greece-replaces-spring-boot/

DEMO

Seeing is believing

Use the compiler of choice

The red pill...

Or the blue pill?

GraalVM™

OpenJDK™

Start to code

code.quarkus.io

```
quarkus ext add kubernetes
```

TAKEAWAYS

Empower local setup with Dev Services
Use Dev Services to achieve a consistent local setup across team members.

Immediate validation with Continuous Testing
Enable Continuous Testing to get instant feedback on local code changes.

Test Kubernetes YAML with *quarkus-test-kubernetes-client*
Validate your Kubernetes resources by testing their YAML content.

Scale up and down quickly with small container images
Create smaller native container images with fast startup.

Deploy smoothly through binding YAML fragments and application configurations
Set up application profiles to share Kubernetes configurations and fragments responsibly across deployments.

The code

https://github.com/ammbra/fellowship

More to learn from

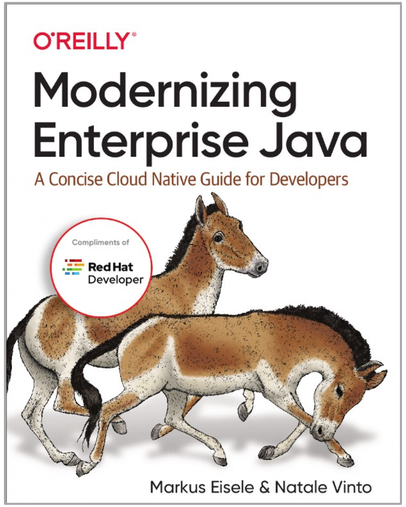
Additional resources

QUARKUS

https://quarkus.io

https://quarkusio.zulipchat.com

@quarkusio



O'REILLY

Modernizing Enterprise Java

A Concise Cloud Native Guide for Developers

Compliments of Red Hat Developer

Markus Eisele & Natale Vinto

Modernizing Enterprise Java

[Download]()

Quarkus Cheat Sheet

[Download]()

Thank you

Red Hat is the world's leading provider of

enterprise open source software solutions.

Award-winning support, training, and consulting

services make

Red Hat a trusted adviser to the Fortune 500.

linkedin.com/showcase/red-hat-developer

youtube.com/redhatdevelopers

facebook.com/redhatdeveloperprogram

twitter.com/rhdevelopers