



QUARKUS

QUARKUS WORLD TOUR 2022

INTRODUCTION TO QUARKUS

WHO'S ON STAGE TODAY?

YOANN RODIETRE



QUARKUS



- Hibernate team at Red Hat
- Hibernate Search lead developer
- Contributor to
 - Hibernate (Search, ORM, Validator)
 - Quarkus (Search, ORM, Validator, Core)
 - Whatever gets in the way...



QUARKUS

SUPERSONIC, SUBATOMIC JAVA

Quarkus World Tour 2022 {👉}

DEMO



<https://code.quarkus.io/?e=resteasy-reactive&e=hibernate-orm-panache&e=jdbc-postgresql&e=smallrye-openapi>

(Use a desktop browser to download the project...)

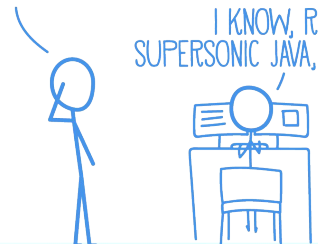
DEVELOPER JOY

A cohesive platform for optimized developer joy:

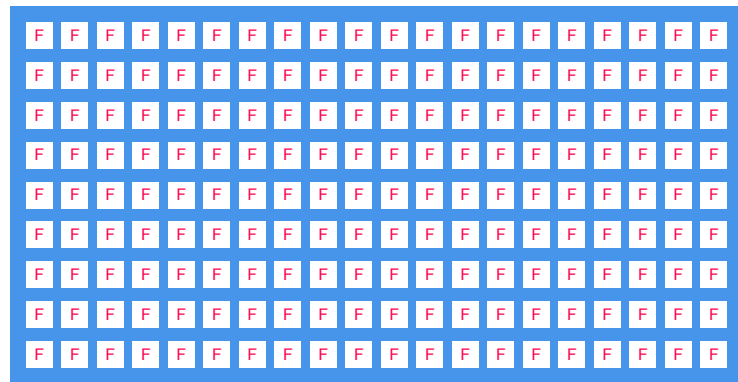
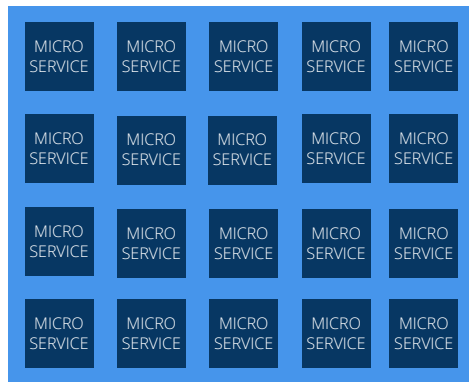
- Zero config, live reload in the blink of an eye
- Seamless Development and testing setup with dev services starting databases, SSO, Kafka etc.
- Unified configuration
- Based on standards, but not limited
- Streamlined code for the 80% common usages, flexible for the 20%
- No hassle native executable generation

WAIT,
SO YOU JUST SAVE IT,
AND YOUR CODE IS RUNNING?
AND IT'S JAVA?!

I KNOW, RIGHT?
SUPERSONIC JAVA, FTW!

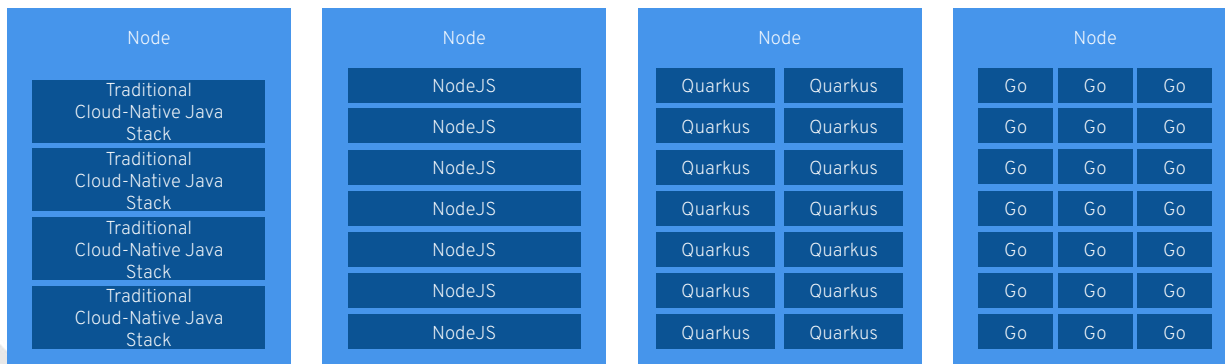


FROM MONOLITH TO...



- 1 monolith \approx 20 microservices \approx 200 functions
- Scale to 1 vs scale to 0
- Start up time

SUPERSONIC SUBATOMIC JAVA



CONTAINER ORCHESTRATION

DEMO



IT'S STILL JAVA!



BEST OF BREED LIBRARIES & STANDARDS



VERT.X

Eclipse
Vert.x



Eclipse MicroProfile



Apache Kafka



Prometheus



Netty



Hibernate



OpenShift



Jaeger



RESTEasy



Apache
Camel

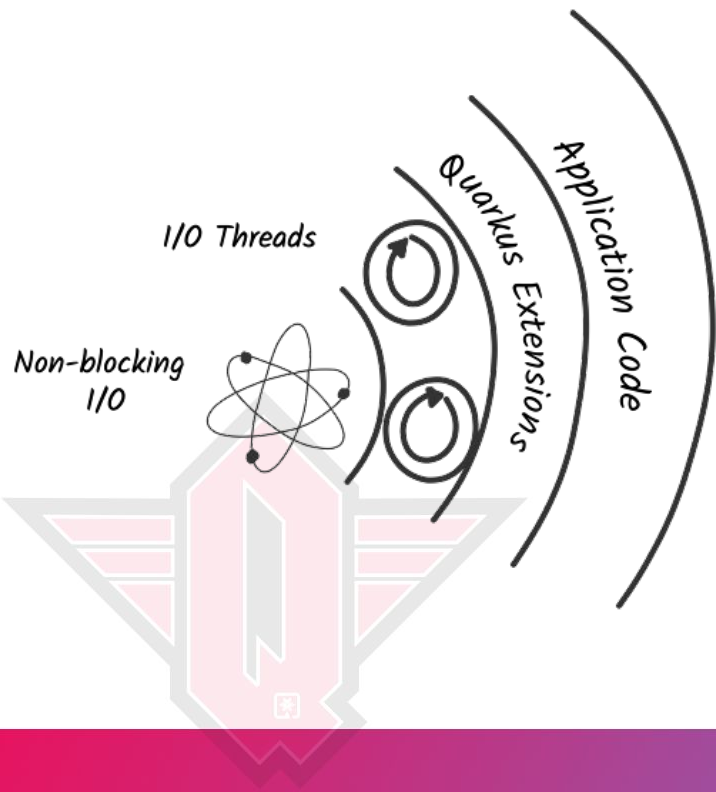


Kubernetes



Spring
Compat

UNIFICATION OF IMPERATIVE AND REACTIVE



```
@Inject  
SayService say;
```

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

```
@Inject @Channel("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
- Use the technology that fits your use-case
- Key for reactive systems based on event driven apps

QUARKUS BENEFITS

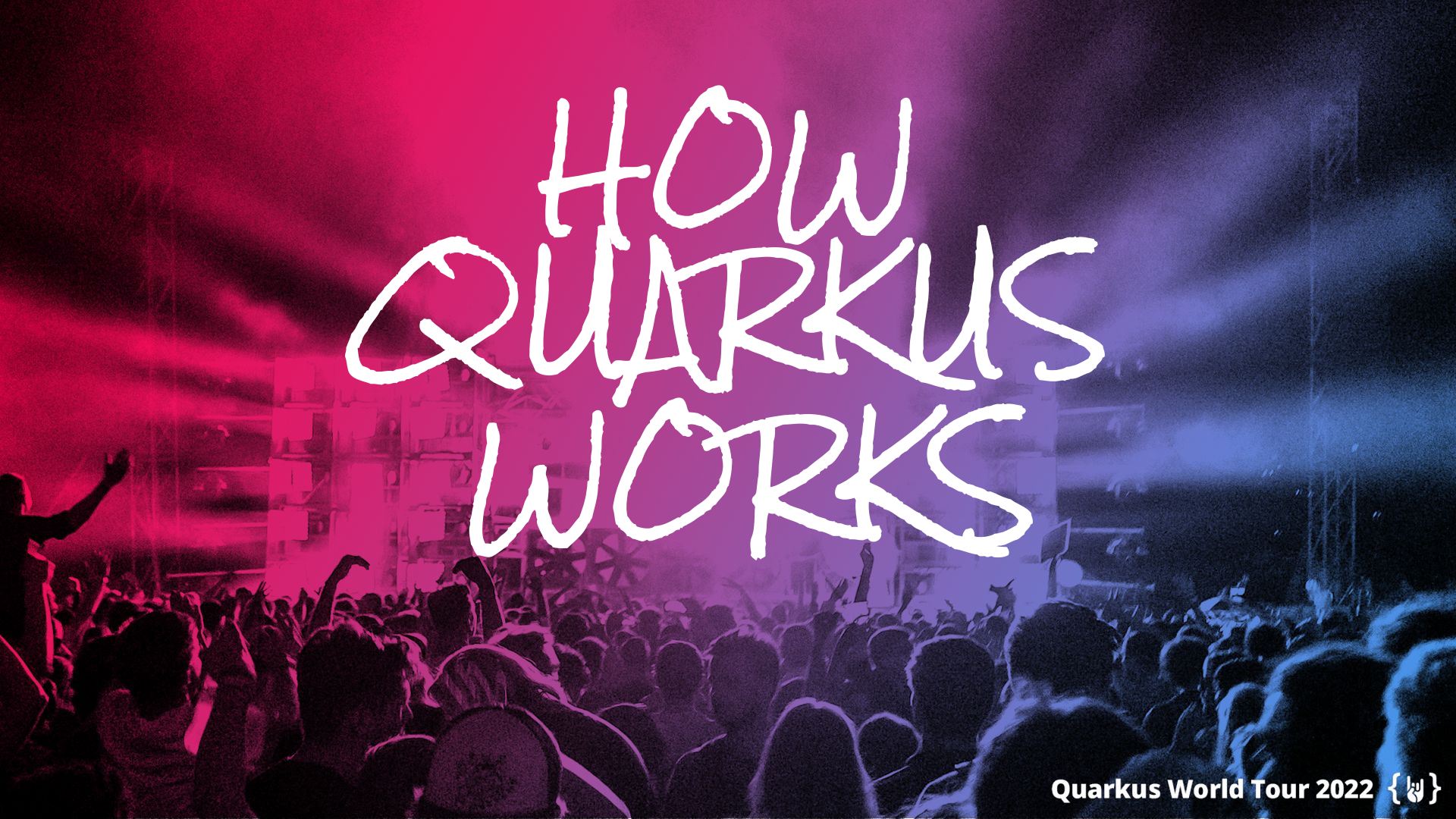
DEVELOPER JOY

SUPERSONIC
SUBATOMIC JAVA

BEST OF BREED
LIBRARIES AND
STANDARDS

UNIFIES
IMPERATIVE AND
REACTIVE





HOW QUARKUS WORKS

Quarkus World Tour 2022 {🎸}

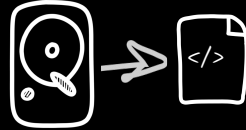
HOW DOES A FRAMEWORK START?

Build Time

Runtime



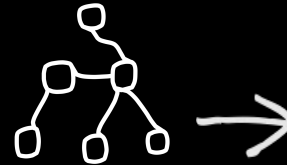
Packaging
(maven,
etc)



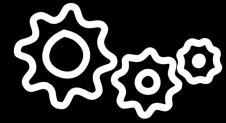
Load config file
from file system
Parse it



Classpath scanning
to find
annotated classes
Attempt to load
class to
enable/disable
features



Build its
model of
the world.

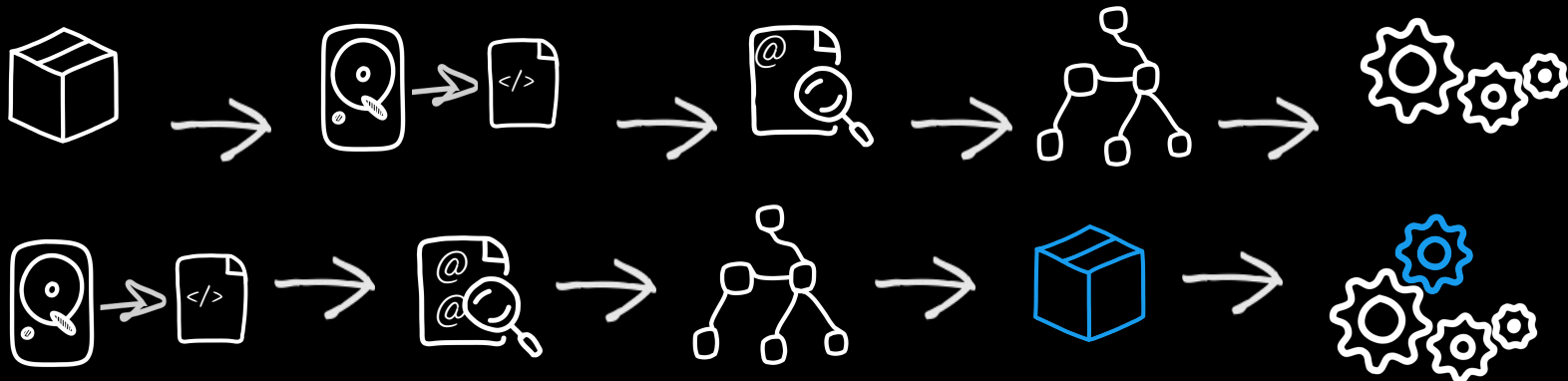


Start the
management
(thread,
pool...)

THE QUARKUS WAY

Build Time

Runtime



Build Time

Runtime

Quarkus Extensions

RESTEasy

Netty

Hibernate

Spring Compat.

MP OpenAPI

MP JWT

Eclipse Vert.X

Agroal (conn pool)

Narayana JTA

MP Reactive
Messaging

Apache Camel

...

Quarkus Core

Jandex

Gizmo

Graal SDK

Arc (DI)

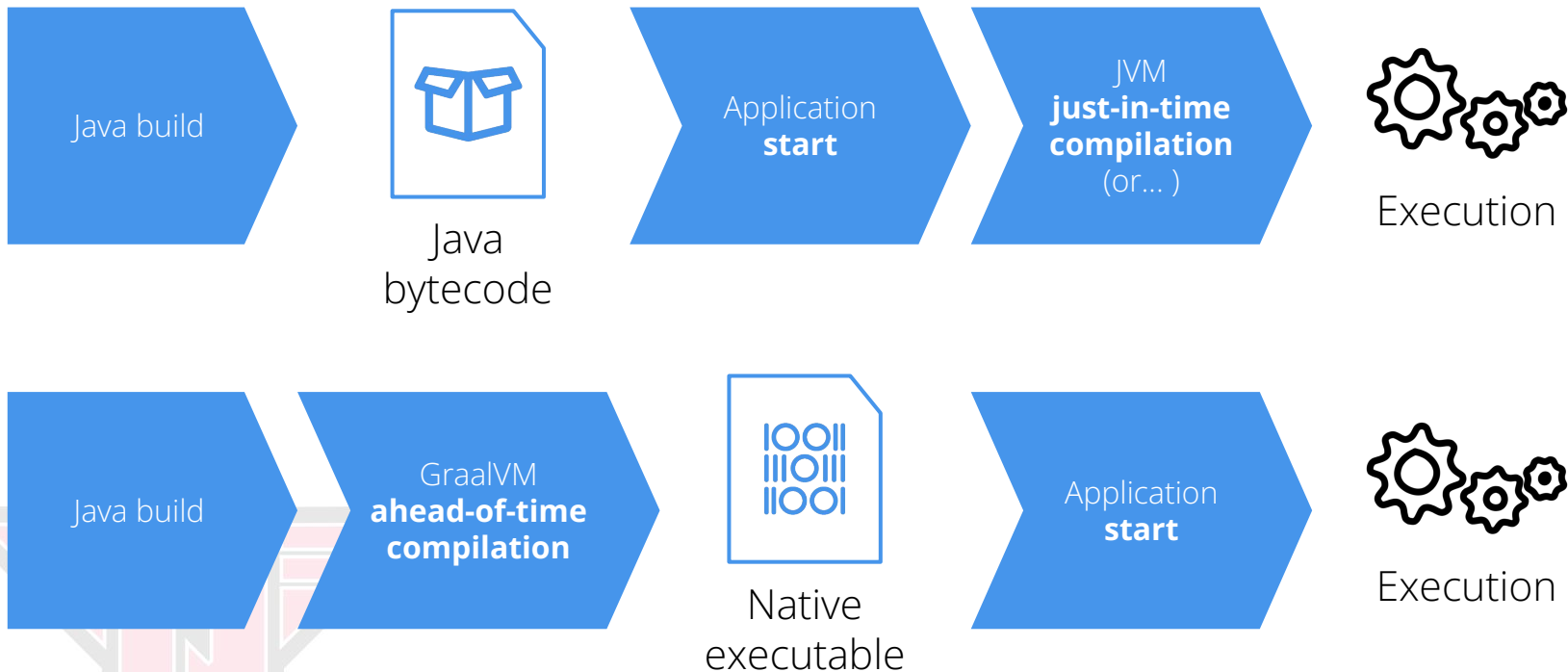
JDK JIT - HotSpot

AOTC - GraalVM Native Image

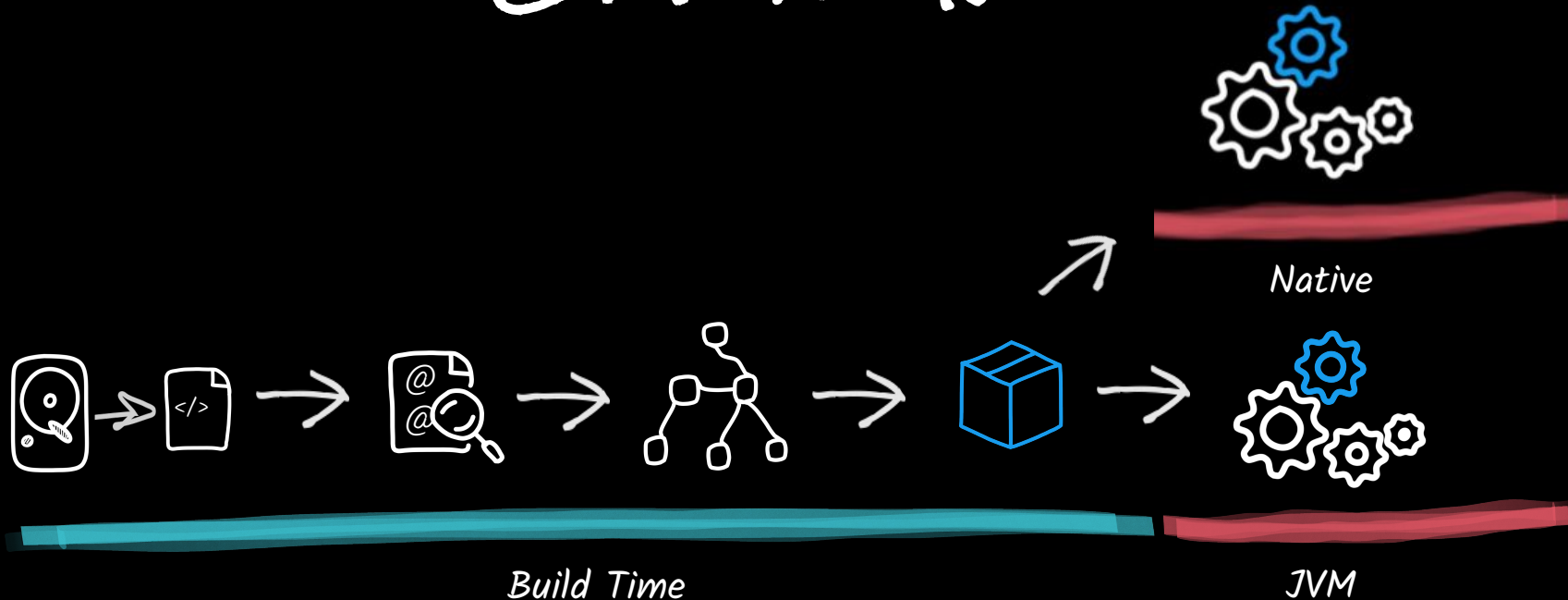
A person wearing glasses and a watch is leaning on a metal railing. The image is overlaid with a red-to-blue gradient. The text "NATIVE COMPILATION" is written in white, hand-drawn style letters across the center.

NATIVE COMPILATION

NATIVE = AHEAD-OF-TIME



THE QUARKUS WAY ENABLES NATIVE COMPILATION



DEMO



THE DARK SIDE OF GRAALVM

NOT SUPPORTED

- **Dynamic classloading**
- InvokeDynamic
- Finalizer
- Security manager
- JVMTI, JMX, native VM Interfaces

OK WITH CAVEATS IN USAGE

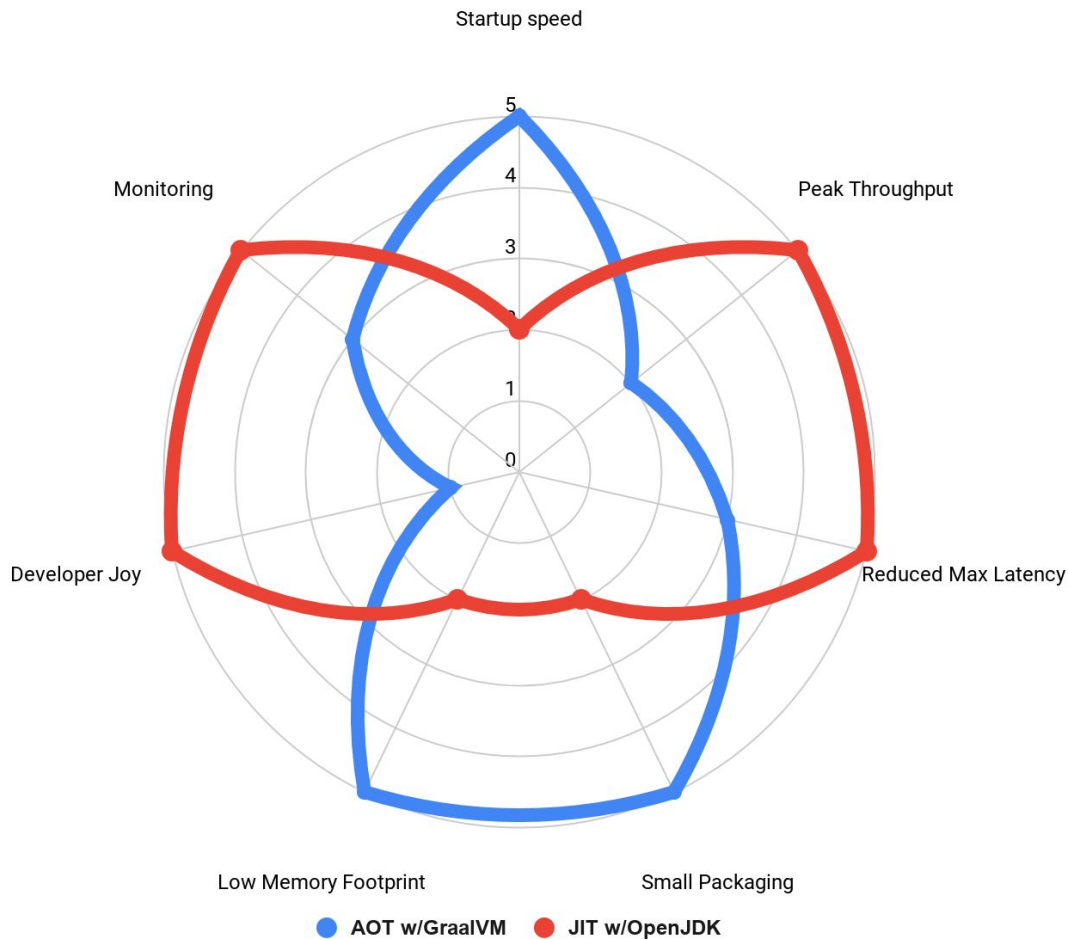
- **Reflection (manual list)**
- Method handles
- **Dynamic proxy (manual list)**
- JNI (manual list)
- **Static initializers (eager)**
- References (similar)

RED: THE "ACTUALLY GOOD" PARTS

GREY: THE "NOT SO BAD" PARTS



JVM VS. NATIVE



NATIVE COMPILATION WITH MANDREL





- Direct downstream distribution of GraalVM
- Near drop-in replacement for GraalVM
- Uses standard OpenJDK instead of GraalVM's custom version
- Focused on native compilation for Java only
- Red Hat contributes upstream!
 - Debugging support
 - ARM support
 - Java Flight Recorder
 - Quarkus part of the upstream GraalVM test suite

USERS OWN STORIES

<https://quarkus.io/blog/tag/user-story/>



DON'T TAKE OUR WORD FOR IT!

 asiakastieto	"We went from 1 min startup times to 400 ms."	RH Press Release Community
 vodafone	"We became increasingly worried about resource consumption that Spring Boot was having while being deployed on the Kubernetes cluster... It became increasingly cumbersome to find ways to circumvent the methodology we were using just to squeeze every little bit of performance out of Spring Boot" - Christos Sotiriou DXL Backend Chapter Lead, Vodafone Greece	Information Week Community
 Lufthansa	"We could run 3x denser deployments without sacrificing availability and response times of service." "Quarkus is close to what our developers are already doing with Spring and it's familiar to them. This is a big benefit"	Community
 wipro	"Before we introduced Quarkus, many of our customers had started to look at alternative stacks like Go and node.js to improve performance and efficiency. These customers were weary of selecting a new language and having to hire new developers or retrain their existing Java developers." - Arijit Mazumdar "There was a low learning curve with Quarkus. It took one of our developers one week to get up to speed on Quarkus and another week to migrate a Spring application to Quarkus." - Arijit Mazumdar "Quarkus and the Spring API compatibility reduced the migration time and complexity which is critically important for our customers." - Arijit Mazumdar	Community

DON'T TAKE OUR WORD FOR IT!



"using Spring with AWS Lambda would have been prohibitive because the startup time of Spring in AWS Lambda is too big from my research" - Dennis Baerten

[Blog](#)

"As costs increase, this is when the benefit of using Quarkus will be experienced due to its more efficient use of cloud resources and fast startup time compared to plain Java and Spring Boot" - Dennis Baerten

"It took me about 3 days to get familiar with the Quarkus stack" - Dennis Berten, Spring Developer



"Some of Payair's developers had mainly Spring experience, we were concerned that it would be difficult for them to "switch sides". It turned out that our fear of the unknown was completely unfounded. Quarkus leverages some good old Jakarta EE standards that all Java developers are familiar with. We did not have to learn a bunch of new APIs." - Hubert Lewandowski

[Blog](#)

"As a long term Spring developer I realized that Spring is slowly becoming the very thing it swore to destroy. The initial premises of Spring (which basically can be summed up as a lightweight alternative to Jakarta EE) are way past the expiry date now. Spring is the undisputed heavyweight champion that can handle everything you imagine but is not your best option for fast and light services. And that applies to Spring Boot as well. - Hubert Lewandowski



"When you adopt Quarkus, you will be productive from day one since you don't really need to learn new technologies." -TalkDesk

[Blog](#)



"After deploying, we found that Quarkus used about 15% of the CPU, 12% of the memory compared to Spring Boot. So far, we are sticking with Quarkus!" - Sam Dacanay, Lead Software Engineer

[Blog](#)

BUT I ALREADY KNOW SPRING...



WHY MIGRATE?

DEVELOPER JOY

SUPERSONIC
SUBATOMIC JAVA

BEST OF BREED
LIBRARIES AND
STANDARDS

UNIFIES
IMPERATIVE AND
REACTIVE

- What if there isn't an extension for my library?
- GraalVM compilation time / is memory intensive...
- How stable is it? There are lots of issues on GitHub...



STARTING FROM SCRATCH



A screenshot of the Quarkus website's 'Guides - Latest' page. The page features a dark header with the Quarkus logo and navigation links: 'ABOUT', 'LEARN', 'COMMUNITY', and a 'START CODING' button. Below the header, the main content area is titled 'Guides - Latest' and includes a search bar labeled 'Find a Guide' and a dropdown menu for 'Select Guides Version' set to '2.13 - Latest'. The page lists various categories such as 'Getting Started', 'Core', 'Web', 'Data', 'Messaging', 'Security', 'Business Automation', 'Integration', 'Command Line Applications', 'Cloud', 'Observability', 'Serialization', 'Tooling', 'Compatibility', 'Miscellaneous', 'Alternative Languages', 'Writing Extensions', and 'Architecture'. There are also sections for 'Getting Started' with sub-sections like 'Be Guided Through First Application', 'Getting Started with Reactive', 'Building Native Executables', and 'Using our Tooling'. A 'Quarkus Cheat Sheet' section offers a PDF download and a link to more cheat sheets on the Red Hat Developers website.

<https://quarkus.io/guides/>

STARTING THE MIGRATION WITH SPRING COMPATIBILITY EXTENSIONS

```
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.acme.todo.domain.TODOEntity;

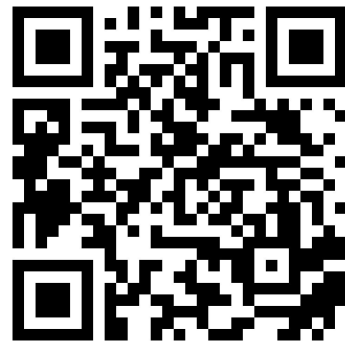
@Repository
public interface TODORepository
    extends JpaRepository<TODOEntity, Long> {
}
```

- Dependency injection
- Spring Web/MVC
- Spring Data JPA
- Spring Data REST
- Spring Security
- Spring Cache
- Spring Boot Configuration Properties
- Spring Scheduled
- Spring Cloud Config Client

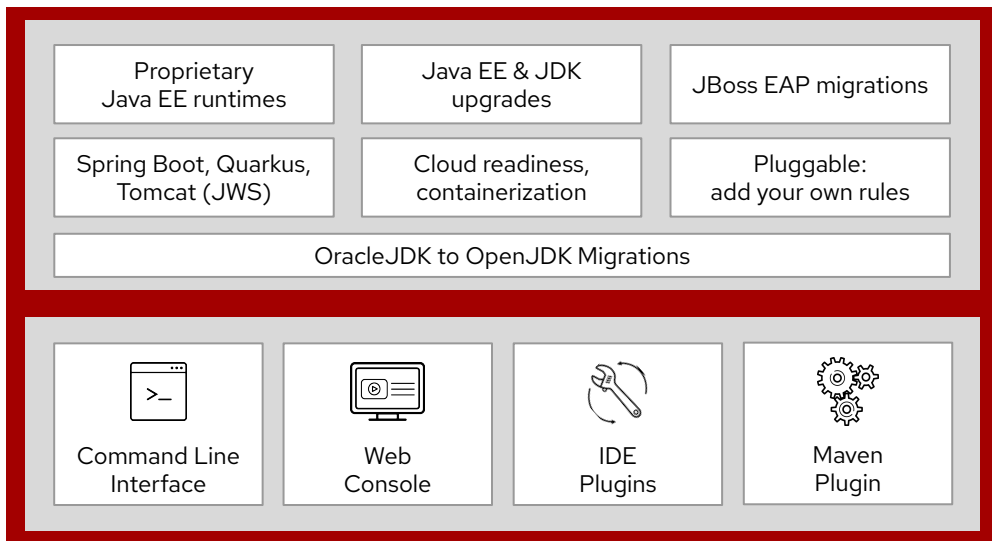
This is intended to help with migration, not for long-term production use.

MIGRATION TOOLKIT FOR APPLICATIONS

“Simplifies the Migration of Spring Apps to Quarkus”



<https://developers.redhat.com/products/mta>



- Automate application analysis
- Estimate level of effort
- Accelerate code transformation & migration
- Includes rules for DI, metrics, security, web, shell, & more

DEMO

<https://red.ht/spring-to-quarkus-todo>





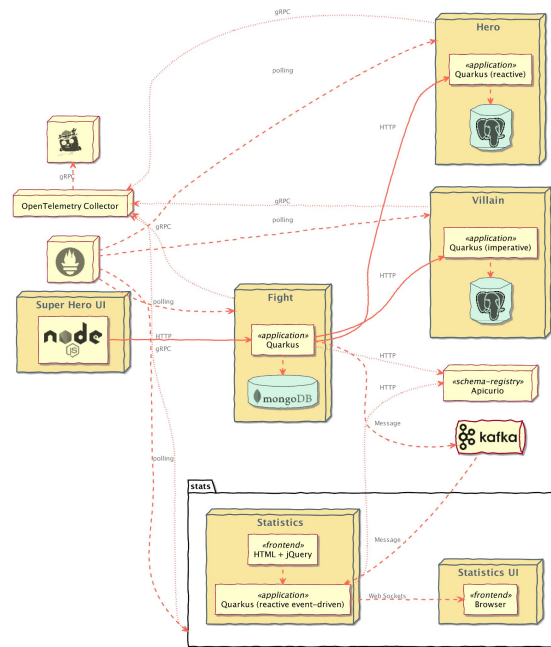
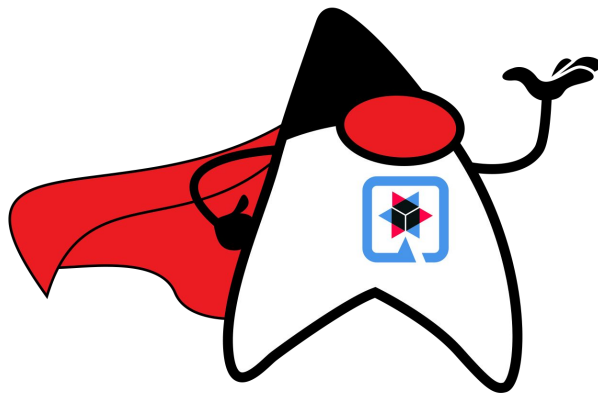
쿼쿠스

Quarkus World Tour 2022 { 쿼쿠스 }



<https://red.ht/quarkus-spring-devs>

QUARKUS SUPERHEROES



<https://github.com/quarkusio/quarkus-super-heroes>

DEVELOPER SANDBOX

- 30 days
- Limited access
- 4 vCPU, 8 GB RAM
- 3 projects
- Cloud IDE, Pipelines



red.ht/dev-sandbox



Add

Select a way to create an Application, component or service from one of the options.

[Details on](#)

Getting started resources

Create applications using samples

Choose a code sample to get started creating an application with.

- [Basic Quarkus](#)
- [Basic Spring Boot](#)

[View all samples](#)

Build with guided documentation

Follow guided documentation to build applications and familiarize yourself with key features.

- [Get started with Quarkus using s2i](#)
- [Get started with Spring](#)

[View all quick starts](#)

Explore new developer features

Explore new features and resources within the developer perspective.

- [Discover certified Helm Charts](#)
- [Start building your application quickly in topology](#)

[What's new in OpenShift](#)

Developer Catalog

All services

Browse the catalog to discover, deploy and connect to services

Database

Browse the catalog to discover database services to add to your Application

Operator Backed

Browse the catalog to discover and deploy operator managed services

Helm Chart

Browse the catalog to discover and install Helm Charts

Managed Services

Discover managed services to simplify deployments and reduce operational overhead & complexities

Git Repository

From Git

Import code from your Git repository to be built and deployed

From Devfile

Import your Devfile from your Git repository to be built and deployed

From Dockerfile

Import your Dockerfile from your Git repository to be built and deployed

Container images

Deploy an existing Image from an Image registry or Image stream tag

Samples

Create an Application from a code sample

From Local Machine

Import YAML

Create resources from their YAML or JSON definitions

Upload JAR file

Upload a JAR file from your local desktop to OpenShift

Quick Starts

Learn how to create, import, and run applications on OpenShift with step-by-step instructions and tasks.

Search:



Binding your Quarkus application to Streams for Apache Kafka

10 minutes

Binding your Quarkus application to the OpenShift Streams for Apache Kafka cloud service

Prerequisites (3)



Get started with Quarkus using a Helm Chart

10 minutes

Deploy a Quarkus application using a Helm Chart.



Get started with Quarkus using s2i

10 minutes

Import a Quarkus Application from git, build, and deploy it onto OpenShift.

Thank you!

Q&A

 <https://quarkus.io>

 <https://quarkusio.zulipchat.com>

 @quarkusio

 <https://youtube.com/c/quarkusio>

 @yoannrodier



<https://yrodier.github.io/presentation/2022-10-06-lorraine-jug-intro-to-quarkus/>

Quarkus World Tour 2022 {👋}