



redhat.



Microsoft Azure

MONOLITHS TO MICROSERVICES: APP TRANSFORMATION

Hands-on Technical Workshop

REACTIVE MICROSERVICES

The 2 faces of Reactive

Actor, Agent
Autonomic
Systems

Reactive
Systems

Akka, **Vert.x**

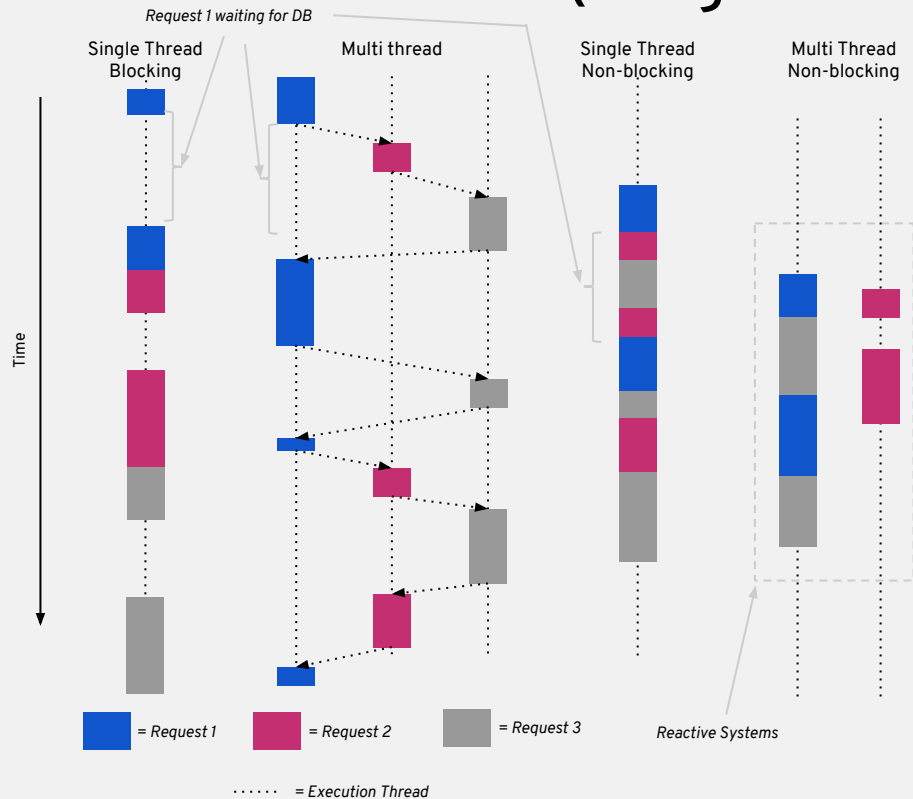
Reactive
**A software showing
responses to stimuli**

Data flow,
Functional
programming

Reactive
Programming

Reactor, RX, **Vert.x**

Execution Model (single core)



Blocking

- Example: CGI, early versions of server side JavaScript.
- Can only scale horizontally

Multi thread

- Example: Java EE, Tomcat, Spring (non reactive)
- Scales horizontally and vertically

Non blocking

- Example: NodeJS, Eclipse Vert.x, Akka, Spring reactive
- Scales horizontally and vertically

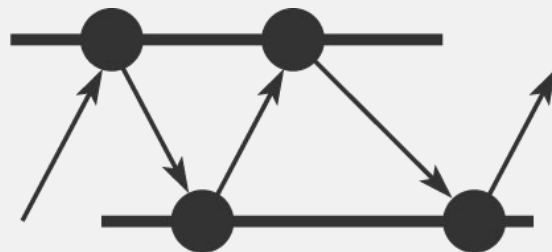
Eclipse Vert.x



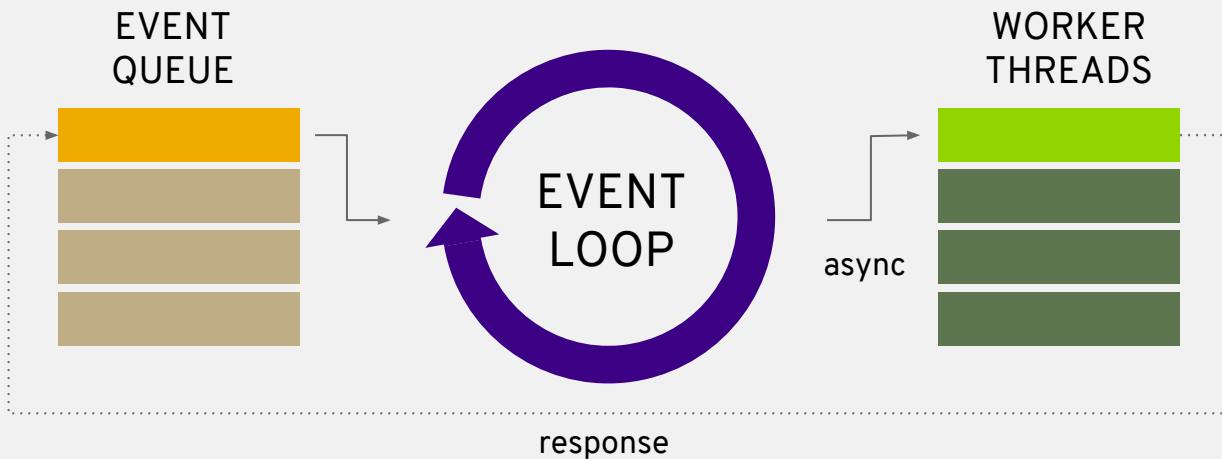
Vert.x is a toolkit to build distributed and reactive systems

- **Asynchronous Non-Blocking development model**
- Simplified concurrency (**event loop**)
- Reactive microservice, Web applications, IOT
- Ideal high-volume, low-latency applications
- Un-opinionated
- Understands clustering in its core architecture

Home - <http://www.vertx.io>

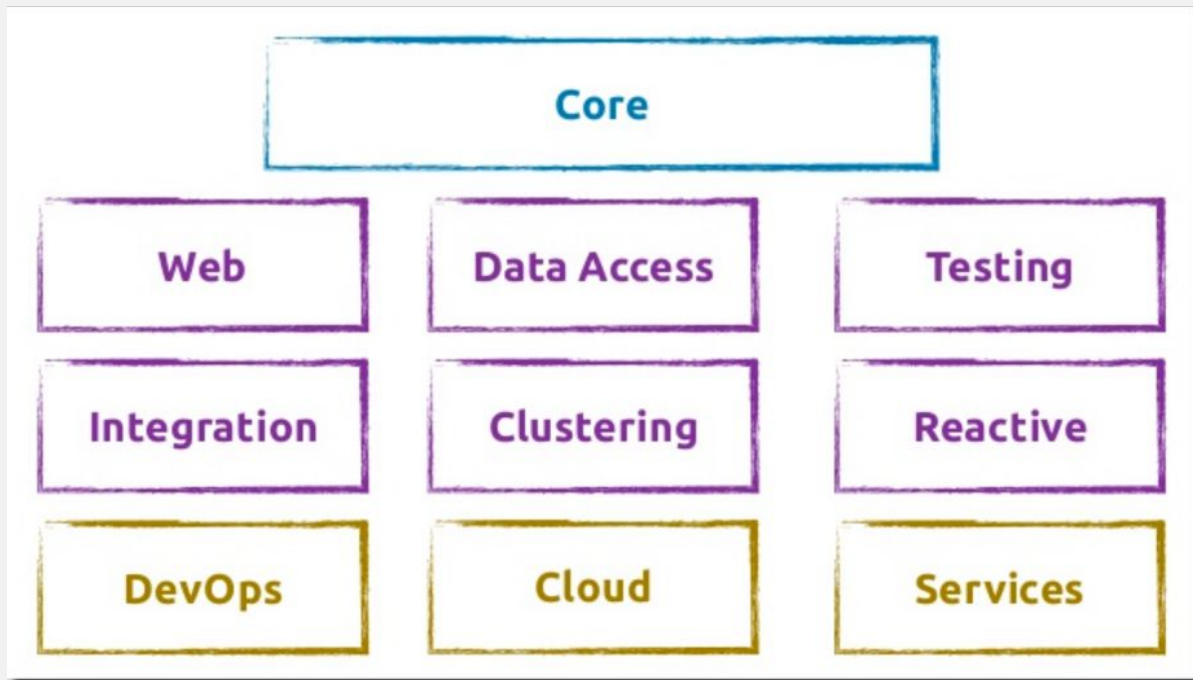


VERT.X EVENT LOOP



Handle Thousands of Requests
With Few Threads

Vert.x Ecosystem



LAB 4: Reactive Microservices with Eclipse Vert.x

- Explore Vert.x Maven project
- Create an API gateway
- Run Vert.x locally
- Deploy Vert.x on OpenShift

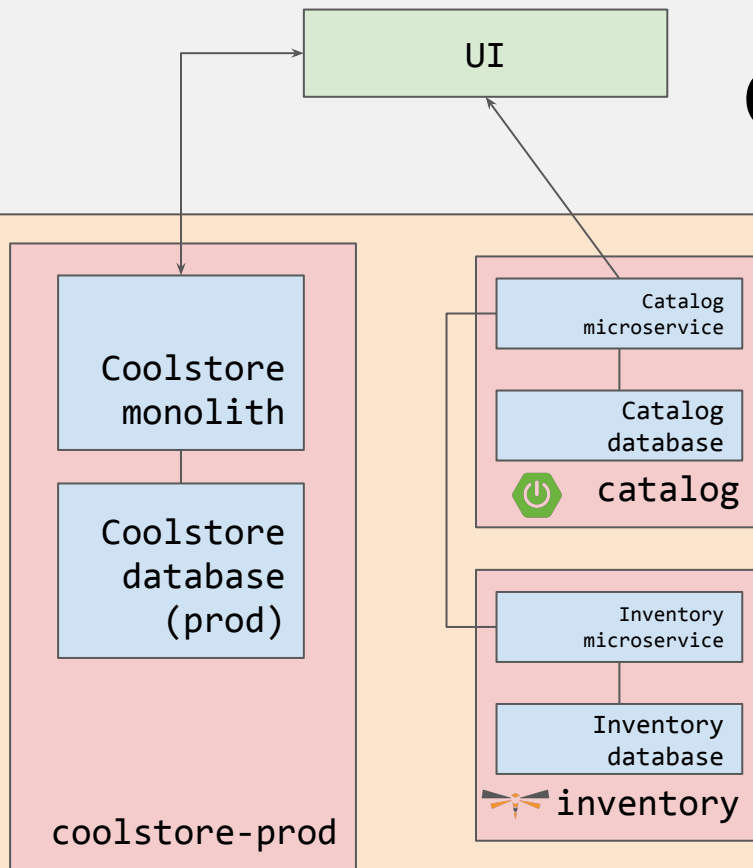
LAB: REACTIVE MICROSERVICES WITH ECLIPSE VERT.X

GOAL FOR LAB

In this lab you will learn:

- How Event-based architectures supercharge microservice apps
- Use cases for reactive applications
- Develop microservices using Eclipse Vert.x
- Interact with other microservices without blocking
- Learn the basics of Reactive programming

CURRENT STATE



Future Services

Future Services

Future Services

Future Services

OpenShift

LAB: REACTIVE MICROSERVICES

A man with wild white hair, wearing a white lab coat and green safety goggles, is holding two pairs of pliers. He is looking intently at the camera. The background is a workshop or lab with various tools and equipment. A dark semi-transparent banner at the top contains the title 'LAB: REACTIVE MICROSERVICES' in white. A similar banner at the bottom contains the text 'SCENARIO 6 BUILDING REACTIVE MICROSERVICES'.

SCENARIO 6

BUILDING REACTIVE MICROSERVICES

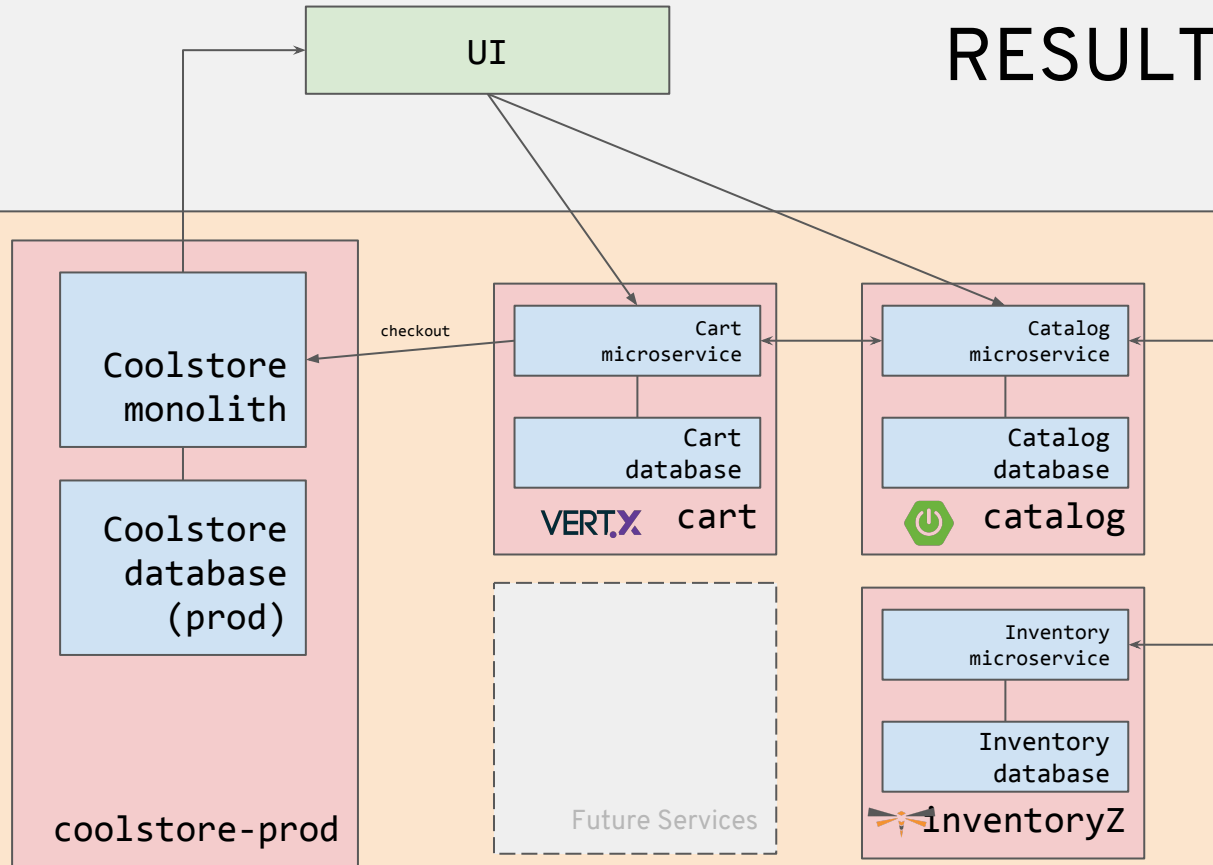
WRAP-UP AND DISCUSSION

RESULT OF LAB

In this lab you learned how to:

- Build reactive web application that are non-blocking
- Asynchronously call out to external service using Callbacks, Handlers and Futures
- Deploy the application to OpenShift

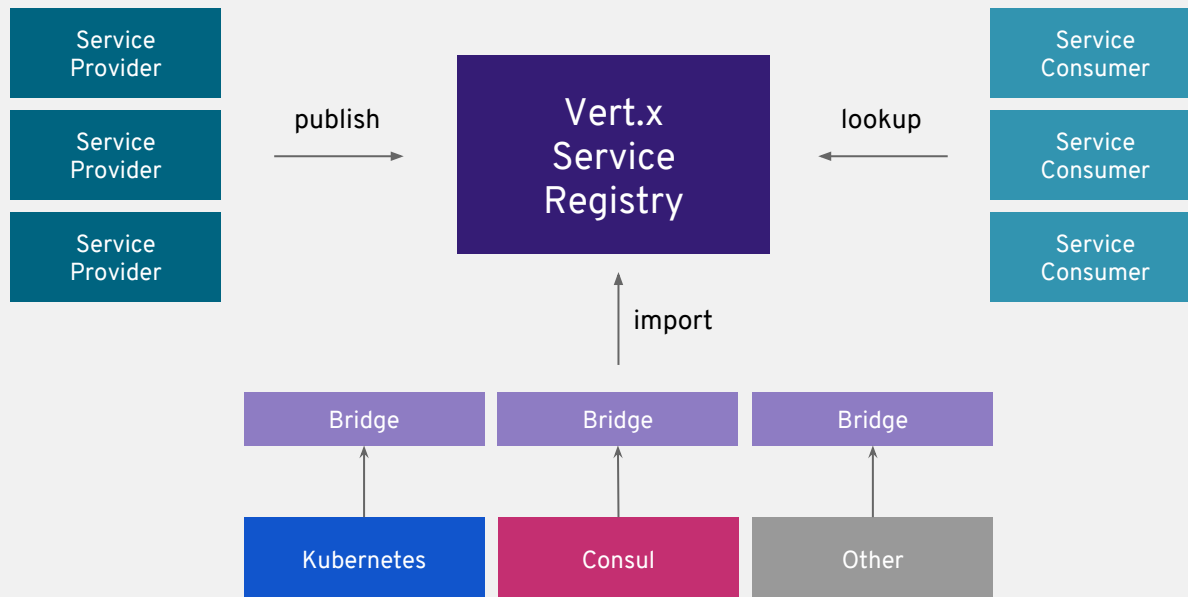
RESULT OF LAB



OpenShift

ECLIPSE VERT.X OFFER MUCH MORE

SERVICE DISCOVERY



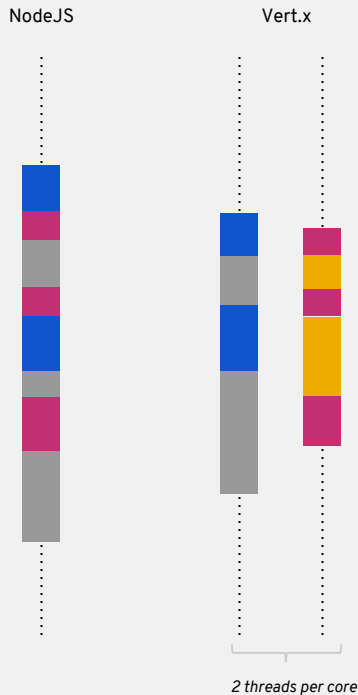
Vert.x vs NodeJS

Vert.x

- Multi-threaded
- Polyglot (Java, JavaScript, Scala, and more)
- Supports reactive programming using RxJava, RxJS, etc

NodeJS

- Single threaded
- JavaScript only
- Support reactive programming using RxJS



FREE E-BOOKS

A gentle **guide** to
asynchronous programming
with **Eclipse Vert.x**
for **Java developers**

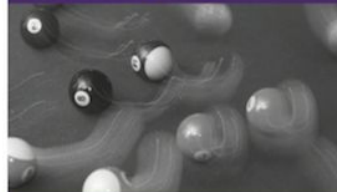
By Julien Ponge, Thomas Segismont & Julien Viet



O'REILLY

Building Reactive
Microservices
in Java

Asynchronous and Event-Based
Application Design



Clement Escoffier

<http://vertx.io/docs/>



THANK YOU



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHatNews



youtube.com/user/RedHatVideos