Introduction

Welcome

During this workshop attendees will have the opportunity to participate in a hands-on Lab, introductory-level session focused on performance monitoring using different Modern cloud-native Java runtimes. Throughout the session, participants will execute tasks such as build, test and deploy microservices using Quarkus (https://quarkus.io), Spring Boot® (https://spring.io) and Micronaut® (https://micronaut.io), on top of the Red Hat® OpenShift® (https://www.redhat.com/en/technologies/cloud-computing/openshift) Application Platform and run load testing scenarios while monitoring the application performance. Specifically, attendees will analyze CPU load and memory usage, establish a baseline, and then observe the impact of scaling up application containers.

This workshop runs in a shared Openshift Cluster hosted on a Red Hat lab environment, but can be easily reproduced in any OpenShift cluster.

What you are going to do:

- Create your own GitHub Repo based on a GitHub Template
- Develop three microservices that consume CPU and memory, each one using a different runtime stack
  - Quarkus
  - Spring Boot
  - Micronaut
- Compile, package and containerize these microservices
- Run these microservices locally on your own Dev Workspace using Red Hat Openshift DevSpaces (https://developers.redhat.com/products/openshift-dev-spaces/overview)
- Build an immutable container image for each microservice using Openshift Pipelines (based on Tekton (https://tekton.dev))
- Deploy your containers to a staging namespace created on Openshift
- Execute and monitor the microservices using different performance parameters
- Load test the microservices
- Analyze CPU load and scale containers appropriately
- Analyze memory and scale containers appropriately
- And much more!

Are you ready to get started? Proceed to the next step to learn more about your environment and get started with the Lab.

Presenting the Workshop

This workshop is about understanding the performance of Java runtimes in a containerized environment in the cloud. It is a hands-on workshop, so you will be doing many things in your development environment and on Red Hat Openshift.

Today, most of us containerize our Java applications and then deploy them to the cloud. But how do we know if our application is performing well? What do we do if we see a CPU or memory usage spike? Do we scale out or scale up? Do we change the JVM parameters? Do we change the container size? And what about having native binaries and compiling our code with GraalVM (https://www.graalvm.org)?

What Is This Workshop About?

To understand performance in a containerized environment we need a few simple algorithms that consume CPU and memory. We will then develop these same algorithms in Quarkus (https://quarkus.io), Spring Boot (https://spring.io/projects/spring-boot) and Micronaut (https://micronaut.io), deploy them to the Red Hat® OpenShift® (https://www.redhat.com/en/technologies/cloud-computing/openshift) Application Platform and monitor their performance.

The algorithm consuming CPU will be a simple loop. The higher the numner of iterations, the more CPU it consumes:

JAVA

```
while (iterations > 0) {
    if (iterations % 20000 == 0) {
        try {
            Thread.sleep(20);
        } catch (InterruptedException ie) {
        }
    }
    iterations--;
}
```