

UPGRADE TO THE CLOUD

---

**STEM 5**

### UPGRADE TO STEM 5

- ▶ OSGI artifacts work in an OSGI container.
- ▶ There is multiple implementation like: Equinox, Apache Felix..
- ▶ But the challenge is to run these artifacts in JEE container keeping their OSGI nature.
- ▶ Then embed the so called JEE-OSGI in an micro service environment
- ▶ Wrap the application in a container to run on any cloud provider

### HOW DO WE COMMUNICATE WITH STEM ARTIFACTS?

- ▶ Eclipse technology offers a core solution to make it possible to expose services as an extension points, using Equinox Bridge Servlet.
- ▶ The problem was in embedding it in a container that supports JEE artifacts like Tomcat.
- ▶ Another problem was to make it also embeddable in a fat jar which is the preferred way to containerize an application for running on the cloud.

## WHAT TECHNOLOGIES COULD SOLVE THE ISSUE?



## ON WORD ABOUT PAPAYA



# DOCKERIZING THE APPLICATION



---

## WHY SPRING BOOT?

- ▶ The open source implementation of Eclipse are very good and have almost but not all of the features of spring boot which are actually the productive features like the configuration server and service registry.
- ▶ When it comes to these two simple but important productive aspects I found spring boot very much ahead.
- ▶ Productive feature of other Solution are sometimes paid.
- ▶ Services offered by Spring boot are provider independent, open source and there are a lot of out of the box solutions are shipped with it.

▶

## STEM 5

---

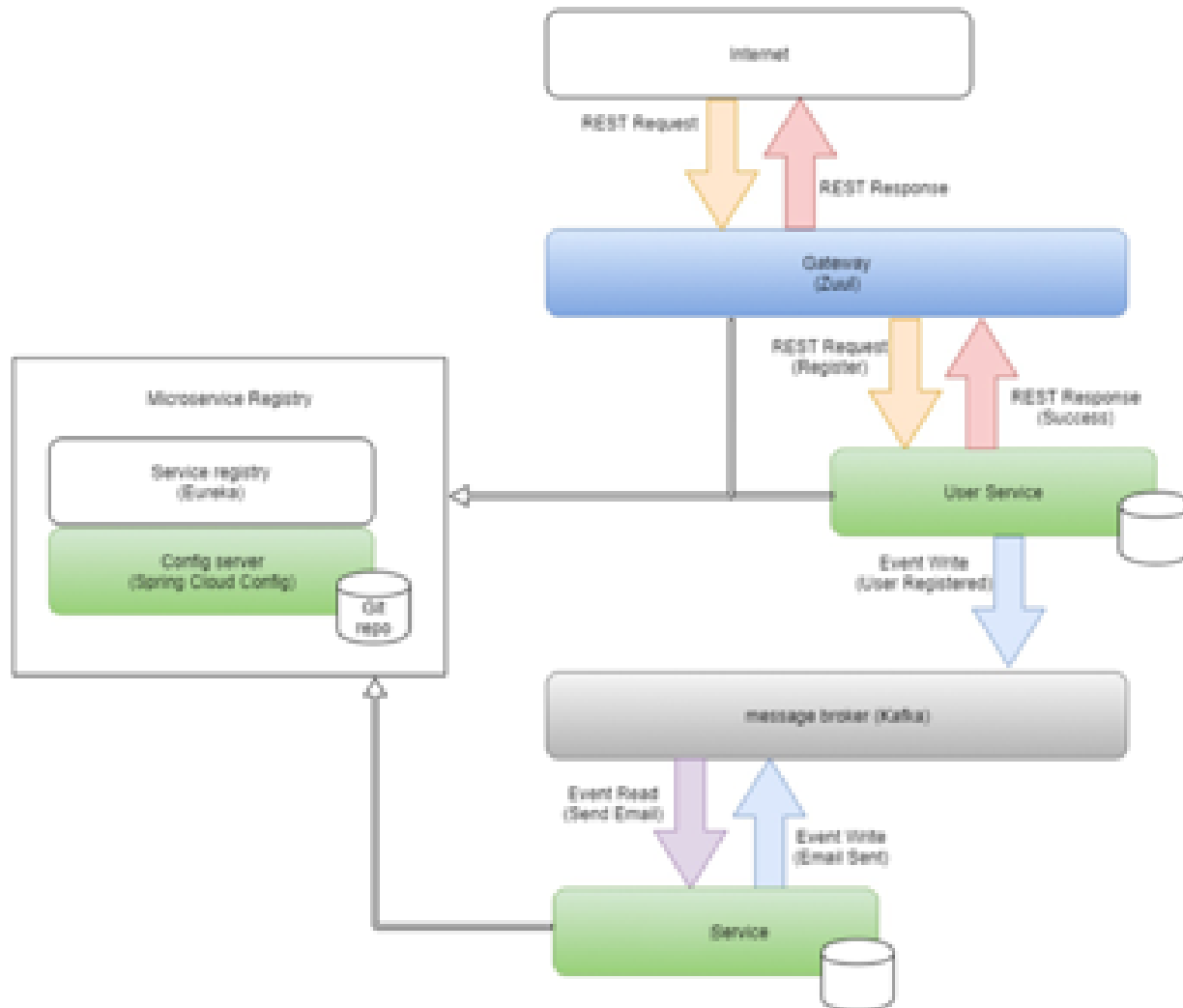
### HOW CLOUD APPLICATION SHOULD WORK?

- ▶ Eureka Server (Service Registry). **DONE**
- ▶ Actuator service **DONE**
- ▶ Configuration Server (Gitlab). **DONE**
- ▶ Hystrix Server (Circuit breaker) **LATER**
- ▶ Authentication Server (keycloak) **DONE**
- ▶ JPA Server (connected to a Casandra database). **In Progress**
- ▶ STEM Core microservice. **STARTED**
- ▶ STEM Model Builder microservice. **STARTED**
- ▶ STEM UI Service (NG). **STARTED In progress.....**
- ▶ Build and CI project **DONE**



# SERVICES DIAGRAM (EXAMPLE)

---



## LOW COHESION (LOWEST)

- ▶ STEM Artifacts are deployed to a container and expose many extension points representing REST API
- ▶ Other applications are calling the services over http requests offered by REST API.

## DEVELOPMENT PLAN

- ▶ Model builder perspective      1 Month
- ▶ Simulation perspective      1-2 Months
- ▶ Analysis perspective      1-2 Months
- ▶ Automatic experiment      1-2 Months
- ▶ Documentation      .....