



OpenTour

Conectando personas y soluciones
para acelerar tu negocio

Madrid

18 de mayo de 2023



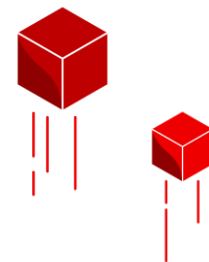
Crash Course desarrollando y desplegando aplicaciones en OpenShift



Ramón Gordillo Ryu
Senior Solution Architect



Carlos Vicens Ken
Senior Solution Architect



Si hay que echar la culpa a alguien ese es Ramón

¿Por qué?

- ▶ Hace unos meses me pidieron ayuda para unos talleres en un cliente
- ▶ Propuse unos cuantos que ya había utilizado anteriormente
- ▶ Extracto de la conversación con Ramón:

R: Venga Carlos, esta gente viene de hacer unos talleres, necesita más... tiene que ser una sesión diferente... más que otro hello world

C: Sujétame el cubata.

When you pray for
rain, you gotta deal
with the mud too

Robert McCall (The equalizer)



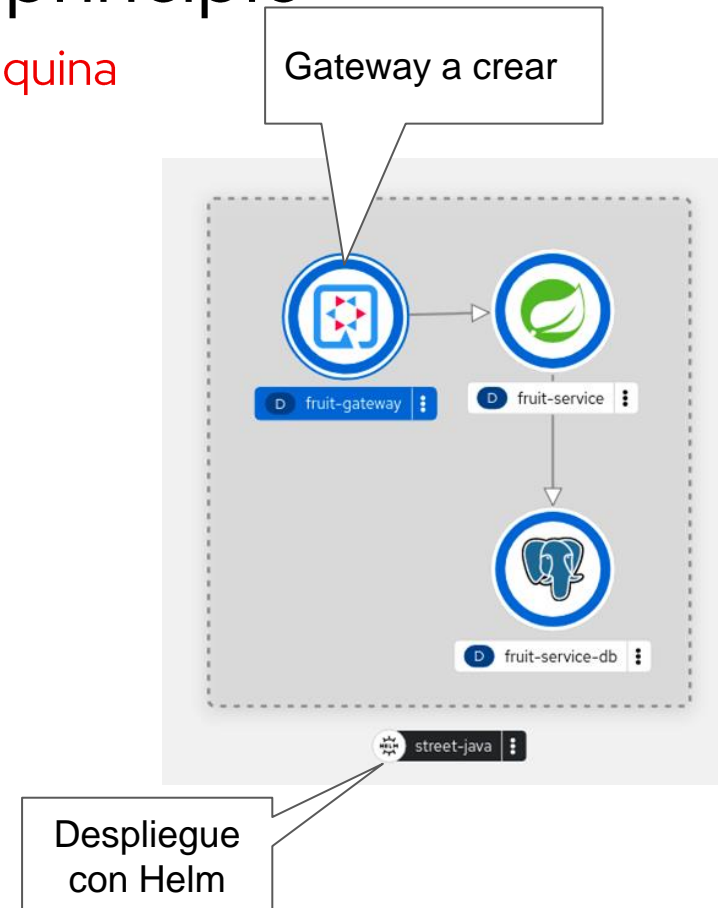
STREET
JAVA



Empecemos por el principio

Que funcione en mi máquina

- ▶ Se nos encarga construir un gateway para exponer un sencillo servicio
- ▶ Parte de la aplicación ya está desplegada en OpenShift utilizando Helm
- ▶ Lo normal es empezar en local, o sea, localhost pero luego habrá que desplegar usando también Helm






Round 1
Fight in localhost

Fight in localhost

- ▶ Arrancamos la aplicación en modo desarrollo en local... utilizamos un cliente rest atacar a nuestro servicio de backend.
- ▶ Podríamos arrancar el servicio en local también... pero este servicio ataca a una base de datos...
- ▶ Igual tenemos acceso a la base de datos directamente... o es una base de datos embebida...

```
mvn quarkus:dev
```

Fight!



Round II
Fight in two places
at the same time

Fight in two places at the same time

Casi como la peli del óscar de 2023

- ▶ Vale, hemos desarrollado ya un parte de nuestro código...
- ▶ Pero no tenemos el backend en local...
- ▶ El otro equipo ya tiene su parte desplegada con un helm chart
- ▶ Además tenemos otro servicio que consumir, Jaeger, para trazas distribuidas, en otro namespace por cierto

Port-forwarding a lo bruto y modificación de /etc/hosts

```
sudo kubefwd svc -n my-project -n street-java-infra
```

```
mvn quarkus:dev -Dquarkus.profile=prod
```

En modo desarrollo pero con perfil de despliegue

Fight!



Round III
Flying kick to OCP

Flying Kick to OCP

Vamos que despleguemos en OpenShift

Se utiliza el registro interno de OpenShift

Extensión k8s/openshift

- ▶ Tenemos algo estable que desplegar en OpenShift... vamos a ello
- ▶ No va a ser fácil... barro
- ▶ Y luego más barro...
- ▶ **Inevitable... necesario... pero barro con el que hay que lidiar**

```
mvn package -DskipTests -Dquarkus.kubernetes.deploy=true
```

```
oc get events --sort-by='.metadata.creationTimestamp' | grep Failed
```

Fight!



Round IV
Hit 'em with the helm!

Hit'em with the helm!

Vamos que en lugar de desplegar a mano utilizamos Helm

- ▶ Nos fijamos en los descriptores generados por la extensión
- ▶ Modificamos helm chart del proyecto para que despliegue también el gateway
- ▶ Antes de desplegar eliminamos los descriptores generados previamente con la extensión

```
oc delete -f ./target/kubernetes/openshift.yml
```

```
helm upgrade \  
  --set fruit-gateway.image.repository=${REPOSITORY} \  
  --set fruit-gateway.image.tag=${TAG} \  
street-java helm/street-java
```

Fight!



Round V
Fighting at the Quay

Fighting at the Quay

Ahora el registro no va a ser local, sino externo

- ▶ Construimos la imagen en local, por sencillez, esto lo haría un pipeline
- ▶ La subimos al registro empresarial, algo que también haría un pipeline
- ▶ Actualizamos la revisión del chart
- ▶ Y... barro nuevamente

```
./fruit-gateway/image-build.sh
```

```
podman push --tls-verify=false \  
  ${REGISTRY}/${USERNAME}/${PROJECT_ID}-${ARTIFACT_ID}:${GIT_HASH}
```

```
helm upgrade \  
  --set fruit-gateway.image.repository=${REG}/${USR}/${PRJ}-${ART}@sha256 \  
  --set fruit-gateway.image.tag=${IMAGE_DIGEST} \  
street-java helm/street-java
```

Fight!



Round VI
Setting ground rules

Setting ground rules

Se trata de reconocer lo que es normal o anormal, primero en localhost

Extensión prometheus

- ▶ Utilizamos el formato prometheus para las métricas, con las que definiremos la línea base
- ▶ Introducimos código para contar errores

```
registry.counter(ACC_ERRORS_COUNT_NAME).increment();
```

- ▶ Leemos métricas en localhost primero

```
mvn quarkus:add-extensions -Dextensions='micrometer-registry-prometheus'
```

```
curl -s http://localhost:8080/q/metrics | grep -i acc_errors_count
```

Fight!



Round VII
**Ground rules also
apply to OCP**

Ground rules also apply to OCP!

Nos llevamos el progreso a OpenShift

- ▶ De nuevo construir imagen, subir al registro
- ▶ Y actualizar el despliegue con una nueva revisión del helm chart
- ▶ Y leemos las métricas pero ahora están en OCP

```
./fruit-gateway/image-build.sh
```

```
podman push --tls-verify=false \  
  ${REGISTRY}/${USERNAME}/${PROJECT_ID}-${ARTIFACT_ID}:${GIT_HASH}
```

```
helm upgrade --set fruit-gateway.image.repository=${REG}/${USR}/${PRJ}-${ART}@sha256 \  
  --set fruit-gateway.image.tag=${IMAGE_DIGEST} \  
  --set fruit-gateway.imagePullSecrets[0].name=street-java-pull-secret \  
street-java helm/street-java
```

```
curl -s https://$(oc get route/fruit-gateway -o \  
  jsonpath='{.spec.host}')/q/metrics | grep -i acc_errors_count
```

Fight!

A pixel art illustration of a city street at night. In the center, a man with a beard and a straw hat holds a large beer. The background features various buildings, including one with a 'PAW' sign and another with a satellite dish. The scene is lit with streetlights and building lights.

Round VIII
Fighting along
Prometheus

Fighting along Prometheus

En OpenShift tenemos un aliado, Prometeo

- ▶ El operador de Prometheus en OpenShift, puede dar servicio a nuestras métricas si se configura correctamente
- ▶ Con un objeto ServiceMonitor indicamos a Prometheus cómo acceder a nuestras métricas

```
apiVersion: monitoring.coreos.com/v1
kind: ServiceMonitor
metadata:
  labels:
    k8s-app: prometheus
  name: fruit-gateway-monitor
  namespace: ${PROJECT_NAME}
spec:
  endpoints:
    - interval: 30s
      path: /q/metrics
      port: http
  namespaceSelector:
    matchNames:
      - ${PROJECT_NAME}
  selector:
    matchLabels:
      app.kubernetes.io/name: fruit-gateway
```

Fight!



Round IX
Sending out an SOS

Sending out an SOS

Prometheus además nos permite definir alertas y enviar notificaciones

```

apiVersion: monitoring.coreos.com/v1
kind: PrometheusRule
metadata:
  name: fruit-gateway-alert-rules
  namespace: ${PROJECT_NAME}
  labels:
    openshift.io/prometheus-rule-evaluation-scope: leaf-prometheus
spec:
  groups:
  - name: fruit-gateway-monitoring
    rules:
    - alert: too-many-accumulated-errors
      annotations:
        description: 'accumulated errors'
        summary: Too many accumulated errors
      expr: acc_errors_count_total > 4
      for: 1m
      labels:
        severity: warning
        special_type: street-java
        alert_type: custom

```

Normal/Anormal



```

---
kind: Secret
apiVersion: v1
metadata:
  name: slack-webhook-secret
stringData:
  slack-webhook-url: ${SLACK_WEBHOOK}
type: Opaque
---
apiVersion: monitoring.coreos.com/v1beta1
kind: AlertmanagerConfig
metadata:
  name: fruit-gateway-routing
  namespace: ${PROJECT_NAME}
spec:
  receivers:
  - name: slack
    slackConfigs:
    - apiURL:
        key: slack-webhook-url
        name: slack-webhook-secret
        channel: '#${SLACK_CHANNEL}'
  route:
    receiver: slack
    repeatInterval: "1m"
    routes:
    - receiver: slack

```

Fight!

¿Pero qué acaba
de pasar?

Miscelánea

Créditos slide 5


- Ken: <https://www.deviantart.com/steamboy33/art/Ken-Third-Strike-HD-291817701>
- Ryu: <https://www.deviantart.com/steamboy33/art/Ryu-THird-Strike-HD-291153215>
- Background: <http://www.vexfx.com/live/tabletop/backgrounds/bgGhostTownNightRain.gif>
- Background: <https://www.deviantart.com/tomtc/art/That-One-Fateful-Night-791554873>

Código y demo


- <https://github.com/atarazana/street-java-on-openshift>

Muchas gracias


Red Hat es el principal proveedor mundial de soluciones empresariales de código abierto con un enfoque impulsado por la comunidad que permite ofrecer tecnologías de alto rendimiento de Linux, nube, contenedor y Kubernetes. Le ayudamos a estandarizar en todos los entornos, a desarrollar aplicaciones nativas de la nube, a integrar, automatizar, asegurar y gestionar entornos complejos gracias al soporte, training y servicios de consultoría galardonados.

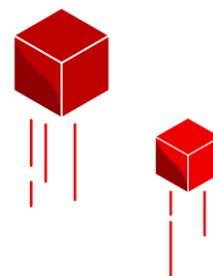
 [linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)

 [facebook.com/redhatinc](https://www.facebook.com/redhatinc)

 [youtube.com/user/RedHatEMEA](https://www.youtube.com/user/RedHatEMEA)

 twitter.com/redhatiberia

 [redhat.com/es/global/espana](https://www.redhat.com/es/global/espana)



When you pray for
rain, you gotta deal
with the mud too

Robert McCall (The equalizer)

STREET JAVA



When you pray for
rain, you gotta deal
with the mud too

Robert McCall (The equalizer)

STREET JAVA



When you pray for
rain, you gotta deal
with the mud too

Robert McCall (The equalizer)

STREET JAVA



When you pray for
rain, you gotta deal
with the mud too

Robert McCall (The equalizer)



STREET
JAVA

