

Gestión de cargas de trabajo JVM con Eclipse JKube

Víctor Orozco

GuateJUG



Entendamos las cargas de trabajo JVM



Kubernetes - Crear la carga

- Programar y depurar la aplicación
- Empacar la aplicación (.war/.jar)
- Escribir el Dockerfile
- Crear la imagen OCI



Kubernetes - El camino a hacia Kubernetes

- Publicar imagen OCI
- Contenedor (Pod)
- ReplicaSet
- Variable
- Mountpoint
- Secret
- Deployment



Kubernetes - El camino desde Kubernetes al mundo

- SVC
- Ingress
- Healthcheck
- Metrics



Descriptor YAML

Ventajas

1. Fácil de leer
2. Soportado en cualquier editor
3. Común en DevOps
-e.g. Github Action,
Bitbucket
Pipelines-

```
1 ---
2 apiVersion: v1
3 kind: List
4 items:
5 - apiVersion: v1
6   kind: Service
7   metadata:
8     annotations:
9       prometheus.io/path: /metrics
10      jkube.io/git-branch: main
11      prometheus.io/port: "9779"
12      prometheus.io/scrape: "true"
13      jkube.io/git-url: git@github.com:tuxtor/microjakarta10.git
14      jkube.io/git-commit: 164802befa59465eb1208d9841bcdcfce4c43b26
15   labels:
16     app: microjakarta10
17     provider: jkube
18     version: 1.0-SNAPSHOT
19     group: com.vorozco
20     name: microjakarta10
```

Descriptor YAML

Desventajas

1. Difícil de mantener
2. No hay especificación
3. Sin soporte a entornos y parámetros

```
1 ---
2 apiVersion: v1
3 kind: List
4 items:
5 - apiVersion: v1
6   kind: Service
7   metadata:
8     annotations:
9       prometheus.io/path: /metrics
10      jkube.io/git-branch: main
11      prometheus.io/port: "9779"
12      prometheus.io/scrape: "true"
13      jkube.io/git-url: git@github.com:tuxtor/microjakarta10.git
14      jkube.io/git-commit: 164802befa59465eb1208d9841bcdcfce4c43b26
15   labels:
16     app: microjakarta10
17     provider: jkube
18     version: 1.0-SNAPSHOT
19     group: com.vorozco
20     name: microjakarta10
```

Helm

1. + Aplicaciones como paquetes
2. + Parametrizable
3. - Proyecto independiente
4. - Crear un chart es complejo



The
package manager
for Kubernetes

Helm is the best way to find, share,
and use software built for **Kubernetes**.



Kustomize

1. + Plantillas en stack
2. + Versión independiente o nativa
3. + Genera (al vuelo) descriptores a partir de bases
4. - Sigue siendo YAML
5. - Aun así es una tarea adicional

Kubernetes native configuration management

Kustomize introduces a template-free way to customize application configuration that simplifies the use of off-the-shelf applications. Now, built into `kubectl` as `apply -k`.

Install kustomize

Use with kubectl

Kustomization.yaml

```
1 bases:  
2 - ldap  
3 patches  
4 - patch.yaml  
5 --
```

```
1 apiVersion:  
2 kind:  
3 metadata:  
4   name:  
5   label:  
6   app:  
7 spec:  
8   repli:  
9   selec:  
10  mat:  
11  a:  
12  templ:  
13  met:  
14  l:  
15  
16  
17  
18  
19
```



Generadores

1. <https://k8syaml.com/>
2. <https://gimlet.io/k8s-yaml-generator>
3. <https://octopus.com/blog/octopus-kubernetes-yaml-generator>
4. ChatGPT?



Eclipse JKube

- Software Project Management
- Compilación y generación de código
- Reportes
- Documentación
- Estándar de facto



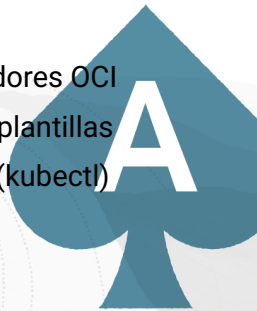
Single source of truth?

Maven

- Nombre de artefactos
- Perfiles y entornos
- Ejecución de tareas (mvn)

Kubernetes descriptor

- Nombre de contenedores OCI
- Perfiles basados en plantillas
- Ejecución de tareas (kubectl)



1. Colección de plugins Maven y Gradle
2. Construcción: Docker, JIB o S2I
3. Despliegue: Kubernetes y/o Openshift



1. Zero
2. Inline (XML)
3. External (Plantillas)



Zero (Spring Boot, Vert.x, Wildfly, Quarkus, Open Liberty, Micronaut)

```
<plugin>  
  <groupId>org.eclipse.jkube</groupId>  
  <artifactId>kubernetes-maven-plugin</artifactId>  
  <version>1.9.1</version>  
</plugin>
```

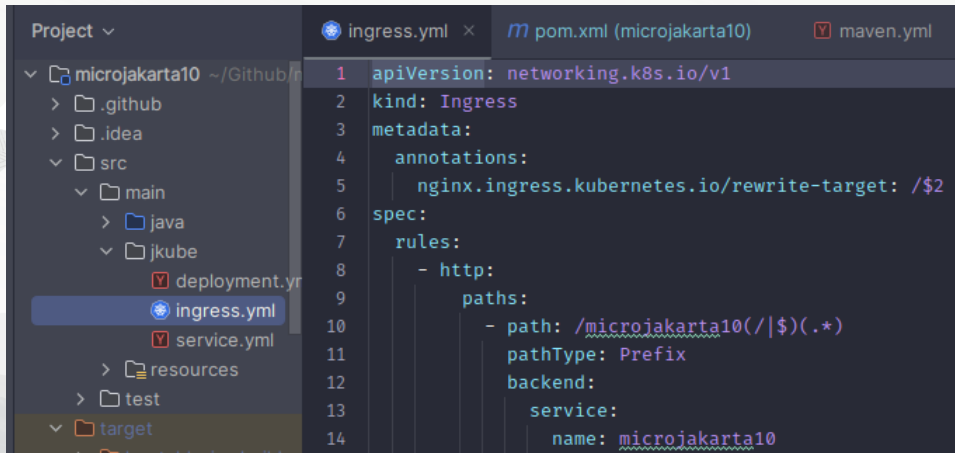


Personalizado con propiedades

```
<maven.compiler.release>17</maven.compiler.release>  
...  
<jkube.generator.name>iad.ocir.io/idemd9jasawg/  
microjakarta10:%l</jkube.generator.name>
```



Ajustado con plantillas



The screenshot shows the Eclipse IDE interface. On the left, the Project Explorer displays the 'microjakarta10' project structure, including folders like '.github', '.idea', 'src', 'main', 'java', 'jkube', 'resources', 'test', and 'target'. The 'ingress.yml' file is selected under the 'jkube' folder. On the right, the editor shows the content of 'ingress.yml' with line numbers 1 through 14. The file is a Kubernetes Ingress manifest. The tabs at the top show 'ingress.yml', 'pom.xml (microjakarta10)', and 'maven.yml'.

```
1 apiVersion: networking.k8s.io/v1
2 kind: Ingress
3 metadata:
4   annotations:
5     nginx.ingress.kubernetes.io/rewrite-target: /$2
6 spec:
7   rules:
8     - http:
9       paths:
10         - path: /microjakarta10(/|$)(.*)
11           pathType: Prefix
12           backend:
13             service:
14               name: microjakarta10
```

Generación de contenedor OCI

```
mvn clean verify k8s:build
```

Generación de descriptor

```
mvn k8s:resource
```

(target/classes/META-INF/jkube/kubernetes.yml)



Despliegue

```
mvn k8s:apply
```

Eliminación

```
mvn k8s:undeploy
```

Logs

```
mvn k8s:log
```

Depuración remota

```
mvn k8s:debug
```



Me gusta por

- Unificación de gestión de proyecto
- Maven profiles
- CI/CD
- Autoconfiguración
- Reducción de errores

Características útiles

- Desarrollo remoto (k8s:remote-dev)
- Despliegue continuo en desarrollo (k8s:watch)
- Helm como alternativa (k8s:helm)





- Víctor Orozco
- me@vorozco.com
- <https://vorozco.com>
- <https://tuxtor.shekalug.org>



Oracle ACE
Pro



This work is licensed under
Creative Commons Attribution-
NonCommercial-ShareAlike 3.0
Guatemala (CC BY-NC-SA 3.0 GT).

