

Iniciación a Openshift Serverless y Knative Serving



serverless



framework



Índice

1. ¿Qué es Openshift Serverless?
2. ¿Qué es Knative Serving?
3. Cómo funciona Openshift Serverless y Knative Serving
4. Caso Práctico
5. Conclusión y Preguntas





1. ¿Qué es OpenShift Serverless?

- Es una plataforma que permite la creación y gestión de aplicaciones sin servidor en la nube.
- Se basa en Kubernetes, lo que significa que los desarrolladores pueden aprovechar las capacidades de orquestación de contenedores de Kubernetes para crear aplicaciones escalables y resistentes.
- Suelen seguir una arquitectura impulsada por eventos para comunicarse entre aplicaciones (no es obligatorio).
- Es más adecuado para aplicaciones asíncronas y sin estado (especialmente las que tienen cargas de trabajo impredecibles).





1.1. Principales ventajas

- Minimizar el uso de recursos.
- Enfoque en el desarrollo.
- Se elimina la planificación de capacidad.
- Provisión automatizada.
- Escalado a cero.



serverless



framework

Iván Piña Castillo



1.2. Uso del Serverless

Cuando SÍ usar Serverless

- Tareas periódicas.
- Procesamiento de IoT.
- Reaccionar a eventos externos.
- Backend de aplicaciones web.
- Procesamiento por lotes.

Cuando NO usar Serverless

- Baja latencia
- Pruebas y depuración fáciles
- Observabilidad





2. ¿Qué es Knative Serving?



- Es una plataforma de código abierto que se ejecuta sobre Kubernetes y que permite la implementación y gestión de aplicaciones sin servidor.
- Proporciona un conjunto de herramientas para automatizar el proceso de implementación, escalar las aplicaciones según la demanda y administrar el tráfico de red.

Se encarga de:

- Desplegar aplicaciones.
- Actualizar aplicaciones.
- Enrutar el tráfico a aplicaciones.
- Escalar automáticamente las aplicaciones.



2.1. Principales ventajas



- Capacidad para simplificar el proceso de implementación.
- Gestión de aplicaciones sin servidor.
- Los desarrolladores pueden centrarse en escribir código de alta calidad y dejar que Knative Serving se encargue del resto.
- Ofrece una gran flexibilidad en cuanto a la elección de lenguajes de programación y frameworks, lo que permite a los desarrolladores utilizar las herramientas que mejor se adapten a sus necesidades.
- La plataforma es altamente escalable y resistente, lo que garantiza que las aplicaciones sin servidor puedan manejar grandes cargas de trabajo.



3. Cómo funciona Openshift Serverless y Knative Serving



- Despliegue de aplicaciones Serverless
- Inspección de aplicaciones Serverless
- Actualización de aplicaciones Serverless
- Gestión de Revisiones de Servicios
- Control de Tráfico
- Configuraciones Auto Escalado





3.1. Despliegue de aplicaciones Serverless



Crearemos un servicio usando:

- Consola web de RHOCP
- Línea de comandos oc
- Línea de comandos Knative (kn)

Podemos crearlo a partir de:

- Un archivo YAML o JAR
- Una imagen de contenedor
- Un repositorio Git





3.2. Inspección de aplicaciones Serverless



Podemos inspeccionar la aplicación usando:

- La Consola Web RHOCP.
- La interfaz de línea de comandos kn.
- La interfaz regular de línea de comandos oc.





3.3. Actualización de aplicaciones Serverless



Podemos actualizar la aplicación:

- Usando el comando "kn service update". (Recomendado)
- Editando el YAML del servicio Knative en la consola web RHOCP.
- Utilizando los comandos "oc patch" o "oc apply".

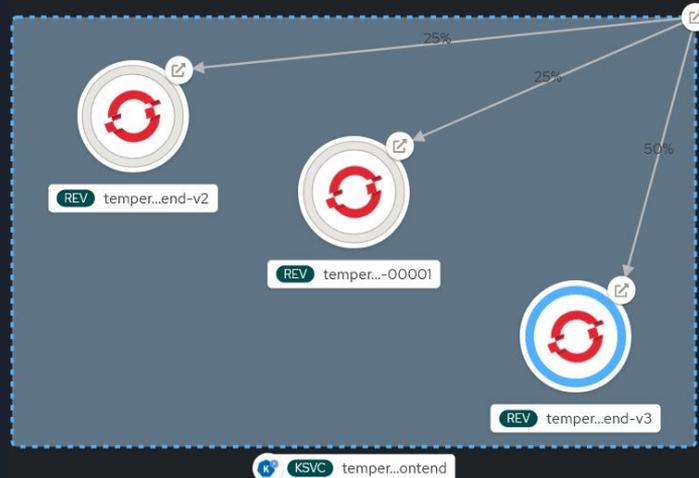




3.4. Gestión de Revisiones de Servicios



- Las revisiones de servicios permiten a los desarrolladores crear múltiples versiones de un servicio y probarlas en paralelo.
- Esto facilita deshacer cambios si es necesario y garantiza que los usuarios no se vean afectados por errores o problemas.





3.5. Control de Tráfico



- Capacidad de controlar el flujo de tráfico hacia diferentes versiones de un servicio.
- Al controlar el tráfico, los desarrolladores pueden asegurarse de que los usuarios sean dirigidos a la versión adecuada de un servicio según factores como la ubicación, el tipo de dispositivo o el comportamiento del usuario.
- Esto mejorará el rendimiento de la aplicación y la experiencia del usuario.



Set traffic distribution

Set traffic distribution for the Revisions of the Knative Service

Split *	Tag	Revision *	
50	actual-version	temperaturas-frontend-v3	⊖
25	first-version	temperaturas-frontend-00001	⊖
25	second-versi ...	temperaturas-frontend-v2	⊖

+ Add Revision

Cancel

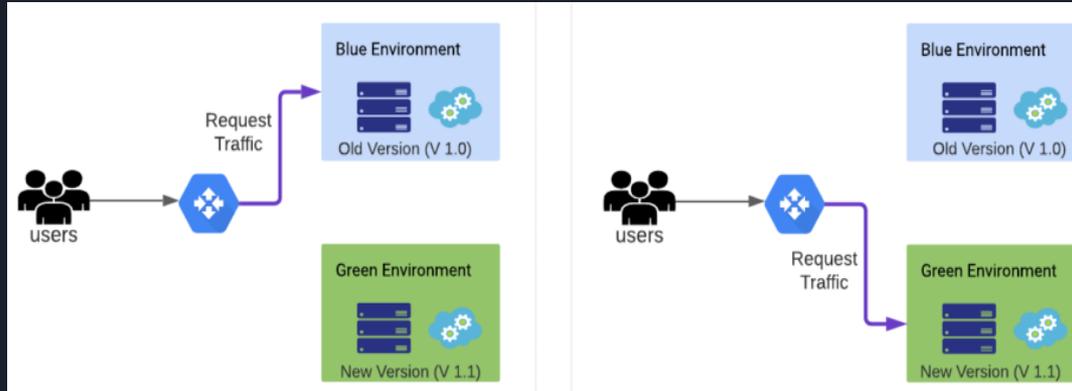
Save



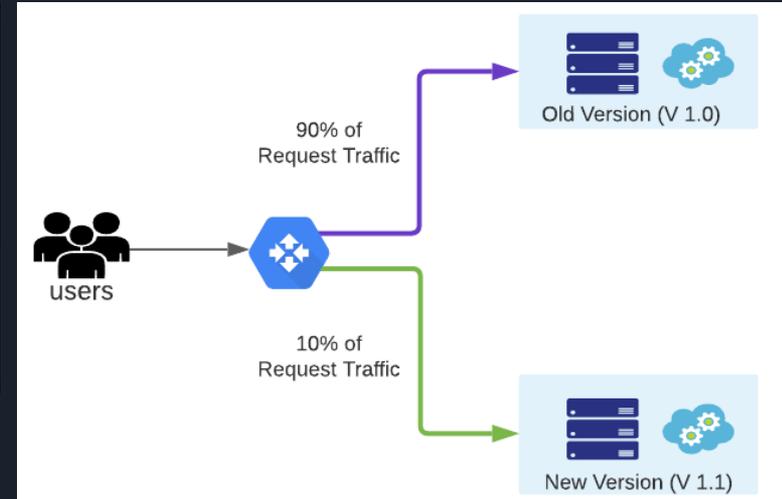
Este control de tráfico flexible permite muchos escenarios de implementación:



Implementación Blue/Green



Implementación Canary





3.6. Configuraciones Auto Escalado



Configuración de auto escalado:

- Escalado a cero:
 - `enable-scale-to-zero`
 - `stable-window`
 - `scale-to-zero-grace-period`

Configuración de límites:

- Escala mínima.
- Escala máxima.

Configuración de concurrencia:

- Límite duro.
- Límite suave.





4. Caso Práctico



DC temper...ackend



REV temper...-00001



KSVC temper...ontend



DC temper...ackend



REV temper...-00001



KSVC temper...ontend





5. Conclusión y Preguntas



El uso de OpenShift Serverless y Knative Serving ofrece numerosos beneficios para el desarrollo y despliegue de aplicaciones en entornos de contenedores.

Estas tecnologías proporcionan una infraestructura flexible y escalable que permite a los equipos de desarrollo aprovechar al máximo los recursos disponibles y optimizar la utilización de la infraestructura en la nube.

