# JBCNConf 2022

# Go vs Java en el contexto de Kubernetes!

Barcelona, Spain

Mauricio Salatino - @Salaboy

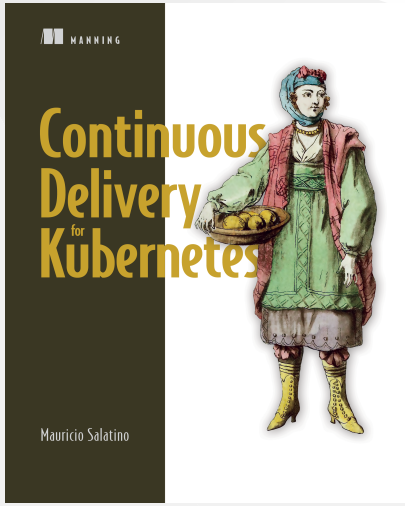
https://github.com/salaboy/from-monolith-to-k8s

# Agenda

- Intro / Background
- IDEs, Lenguajes y Frameworks
- Hablemos de Containers y Kubernetes
- Extendiendo Kubernetes
- Alternativas más saludables

# Intro

@Salaboy



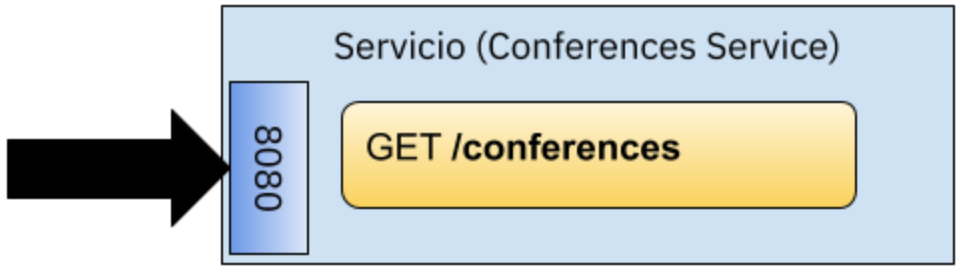

http://mng.bz/jjKP

# Background

- Java / J2EE / Java EE 5
- JBoss
  - Java EE / Wildfly
  - Kubernetes
- Mi primer Kubernetes Controller con Fabric8.io
- Jenkins X
- Spring Boot y Spring Cloud
  - Spring Cloud Kubernetes
  - Kubernetes Controllers con Spring Cloud Kubernetes
- Go
  - Kubernetes Controllers con KubeBuilder
  - Knative Eventing && Knative Functions WG co-lead

# IDEs, Lenguajes y Frameworks

Vamos crear un servicio que expone un endpoint REST




- Goland and Intellij Idea
  - Spring Boot & Quarkus
  - Go

# Resumen Go

- Ventajas
  - Administracion de dependencias (Go Modules) integrada
  - Unit Testing integrado
  - Marshalling de YAML y JSON integrado
- Desventajas
  - No hay frameworks defacto como Spring Boot, cada uno elige e integra
  - Go crea binarios que dependenden de la plataform donde hacemos el build. Similar a los problemas que vamos a tener con GraalVM

# Hablemos de Containers y Kubernetes

Creando containers y YAMLs:

- Spring Boot
  - Eclipse JKube Maven Plugin (ex-Fabric8 Maven Plugin)
    - `mvn k8s:push` / `mvn k8s:resource` / `mvn k8s:apply`
- Quarkus
  - Quarkus Kubernetes Extension
- Go
  - `google/ko`
  - `ko build main.go` / `ko resolve` / `ko apply`

Issue for Spring Boot: https://github.com/spring-projects/spring-boot/issues/31662

# Resumen Go

- Spring Boot y Quarkus proveen integraciones con Jib y **Buildpacks** para contruir containers sin Dockerfiles
    - Ambas integraciones usan la version definida en Maven para taggear el container
- En Go podemos usar Ko para construir y publicar estos containers a nuestro registry preferido
- `ko` construye y publica containers usando un SHA. Nos permite correr containers siempre los ultimos cambios
    - `ko` no crea YAMLs pero `ko resolve -f` reemplaza las referencias
    - `ko apply -f` construye, publica y despliega nueva versiones

Issue for Spring Boot: https://github.com/spring-projects/spring-boot/issues/31662

# Kubernetes APIs

Tarde o temprano vamos a querer interactuar con las APIs de Kubernetes:

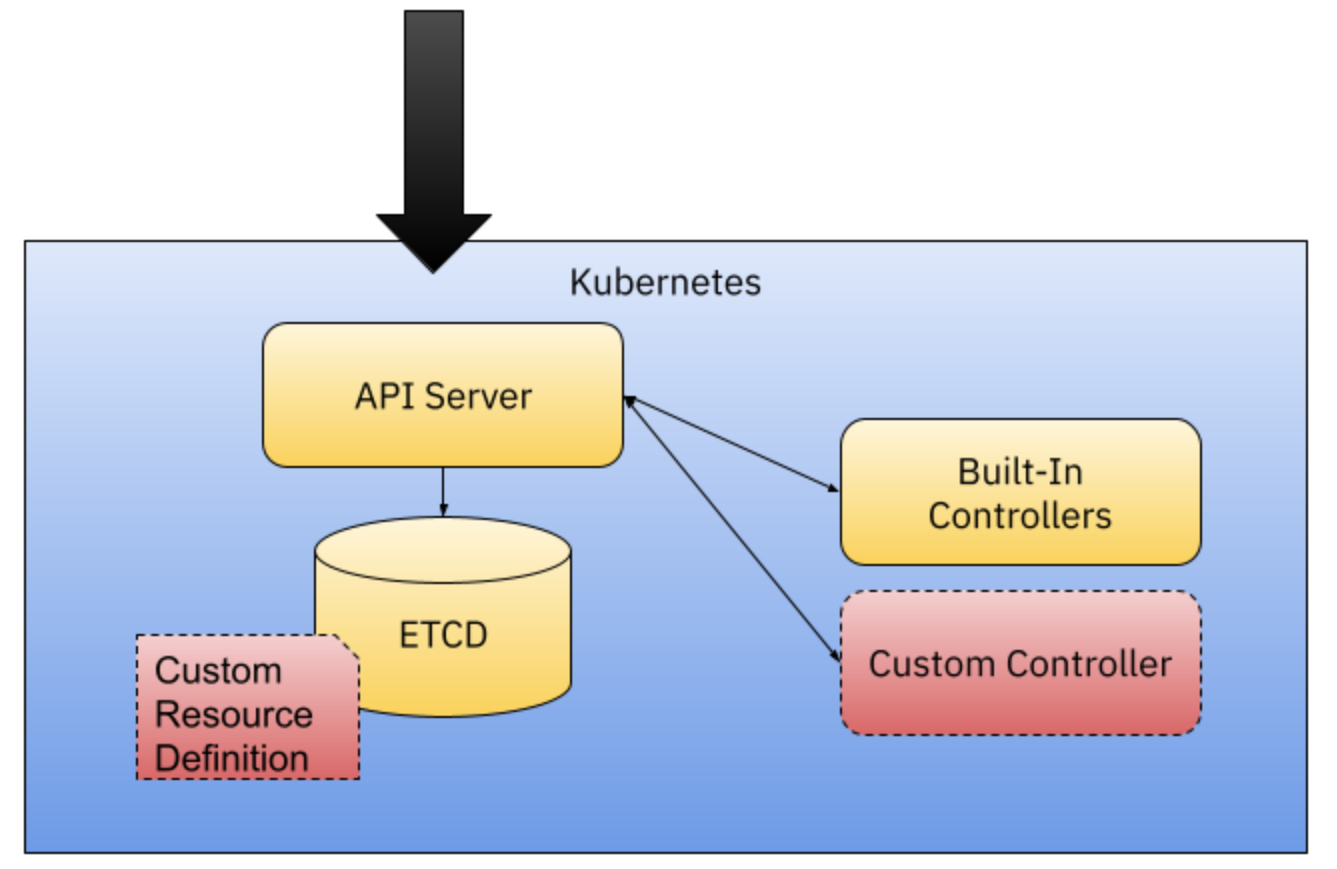
- Fabric8.io Kubernetes APIs
  - Ejemplo
- Kubernetes Client Java
  - Ejemplo
- Go Client
  - Ejemplo

# Cuando extender Kubernetes

- Automatizacion de tareas manipulando recursos de Kubernetes (Controlladores que instalan, configuran o monitorean componentes)
- Necesitamos conceptos de mas alto nivel que Kubernetes no provee
- Integraciones entre distintos proyectos o con servicios externos a Kubernetes

# Como extender Kubernetes

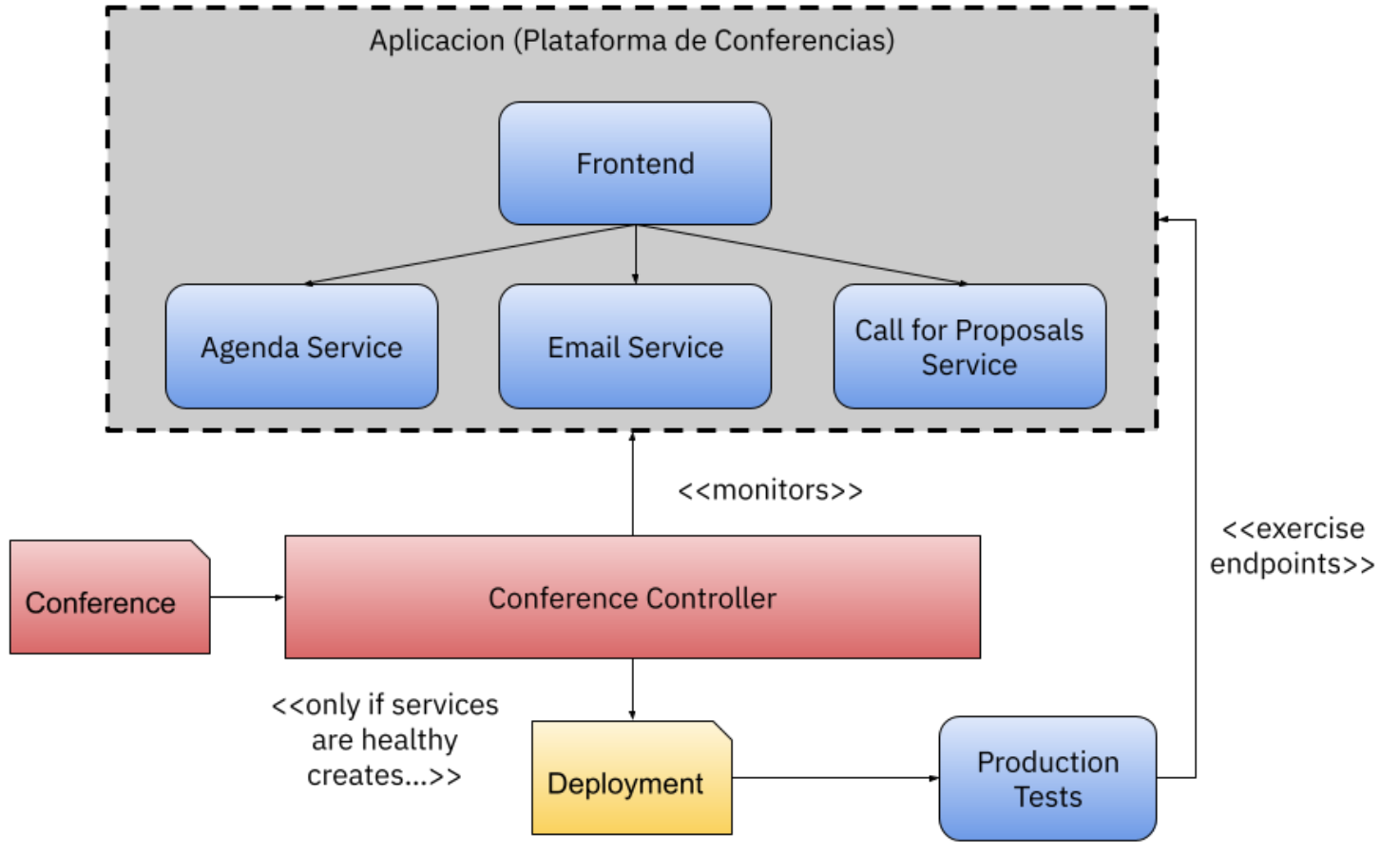
Creamos nuevos Custom Resource Definitions y Kubernetes Controllers




Recomendado Understanding Kubernetes tools/cache package Blog

# Caso de uso de ejemplo

Monitorear y correr test de producción



# Veamos algunas herramientas

- KubeBuilder Go
- Java Operator SDK
  - Bug/Edge Case - FIXED
  - Spring Documentation updated

# Menciones especiales

- Knative Sample Controller: Este Sample Controller usa los mismos mecanismos que usan los controllers de Knative. Estos controllers estan probados en escenarios de alta demanda y han madurado por mas de 4 años. Estos controllers pueden correr multiple replicas concurrentes mirando distintos `buckets` de recursos.
- Kubernetes Client Java: El cliente oficial de Kubernetes Java contiene hoy en dia una version de Controller Runtime, con objetos como Controllers y Reconcilers.
- Go Operators SDK Usa Kubebuilder y provee integraciones con Helm

# Porque no construir K8s Controllers

Pero en 2022 deberiamos escribir controllers?

=> **K8sControllers == EdgeCaseFactories**

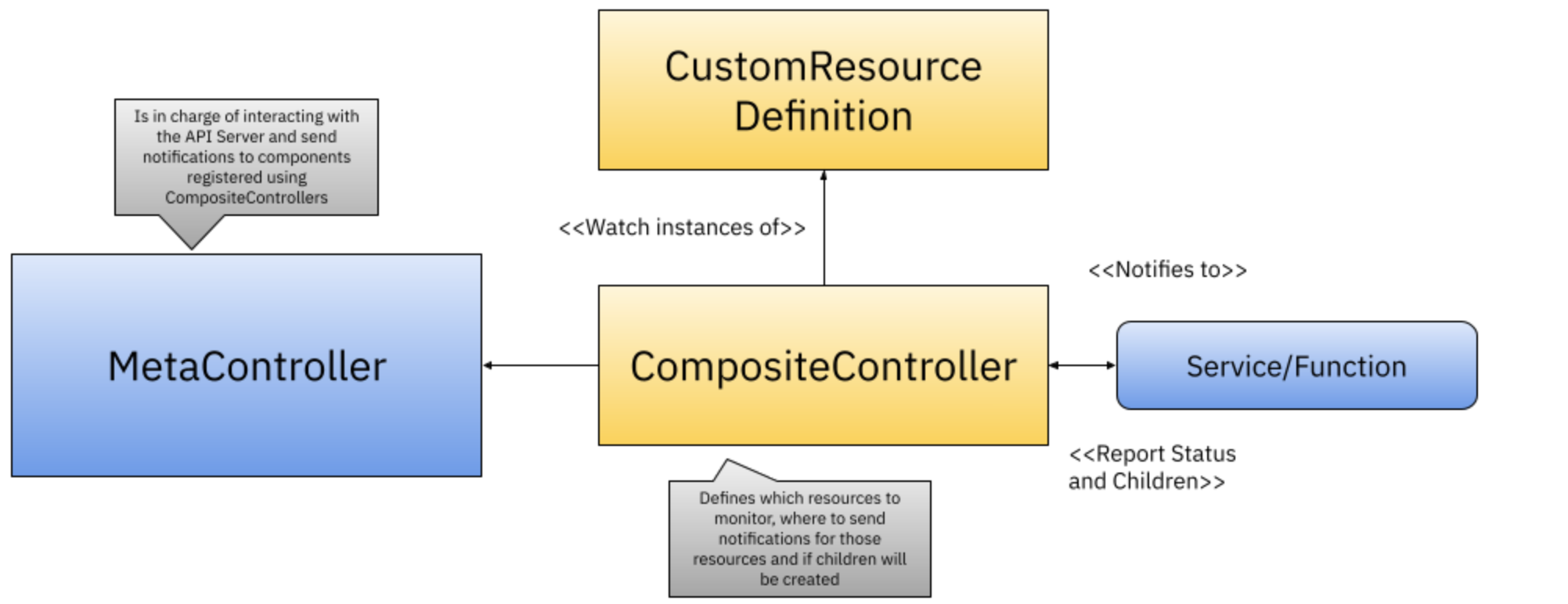
- Requiren permisos especiales para acceder a las APIs de K8s
- Son componentes complejos en un mundo de aplicaciones distribuidas
- Tienen que poder co-existir otros controlladores
- Tienen que escalar siguiendo los lineamientos de Kubernetes / buenos ciudadanos (leader election , alta disponibilidad)

# Alternativas más saludables

- MetaController
- CloudEvents para intregraciones
- Crossplane Providers

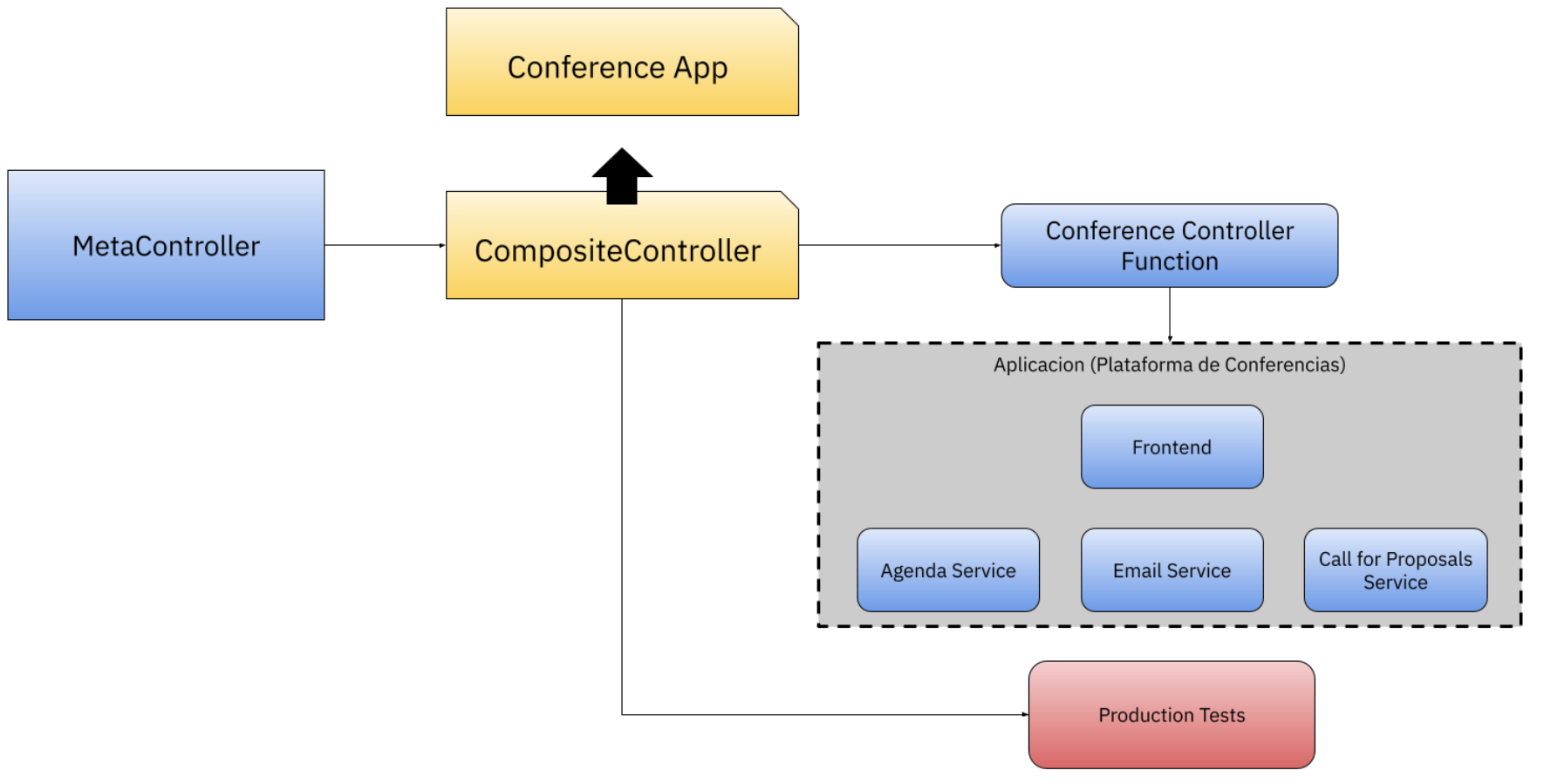
# MetaController

Para no crear y mantener codigo complejo



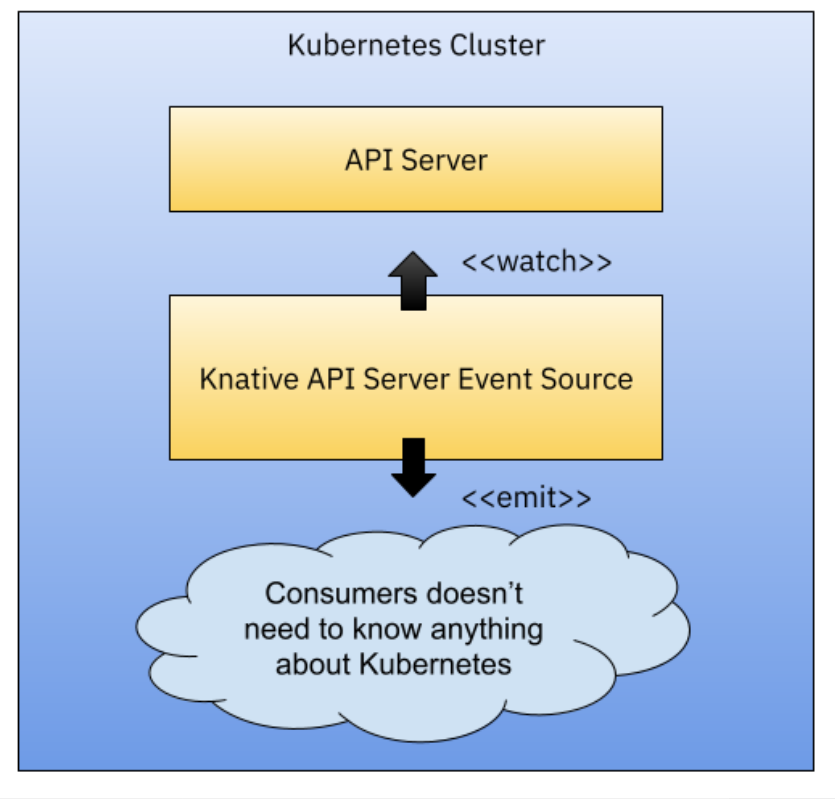

Spring Boot PR

# MetaController Ejemplo



# CloudEvents para intregraciones

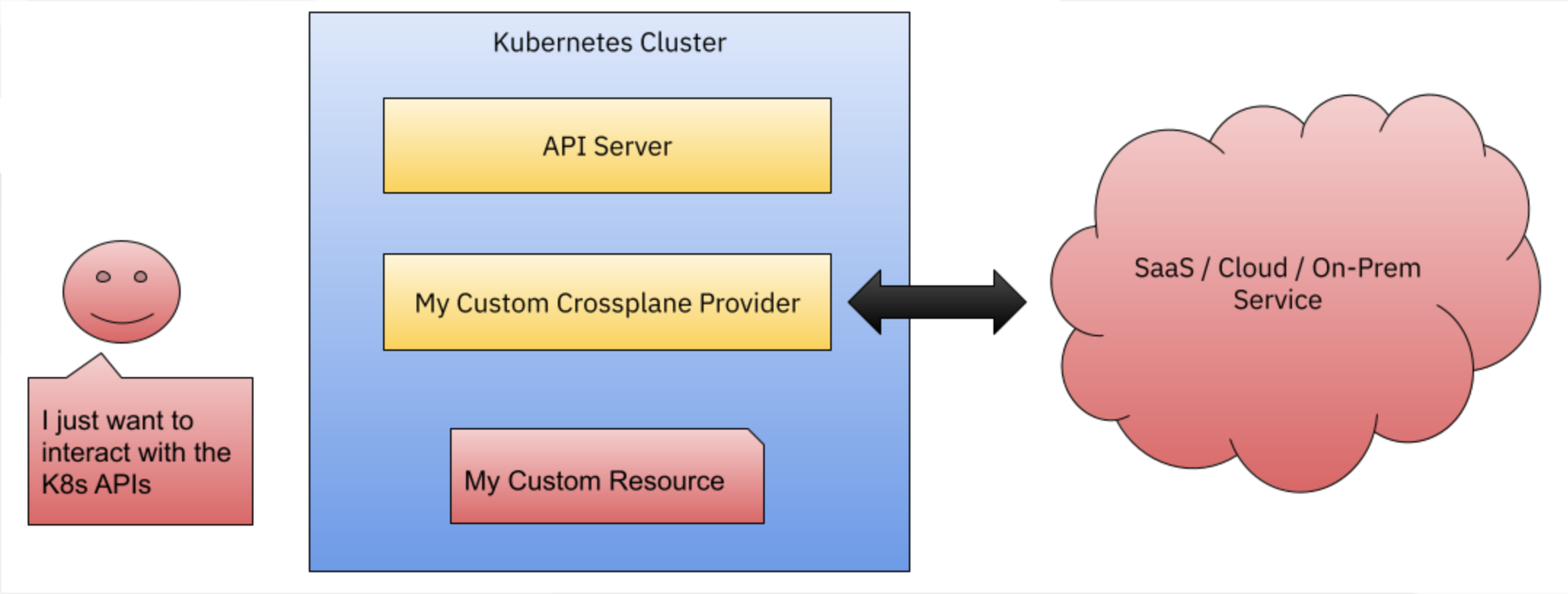
Para escenarios de integracion donde no queremos que todos los componentes tengan accesso a las APIs de Kubernetes




https://knative.dev/docs/eventing/sources/apiserversource/

# Crossplane.io Providers

Para situaciones donde queremos integrar servicios externos a Kubernetes




https://crossplane.io

# Gracias!

Espero haberlos bombardeado con información util!

@Salaboy

Knative