

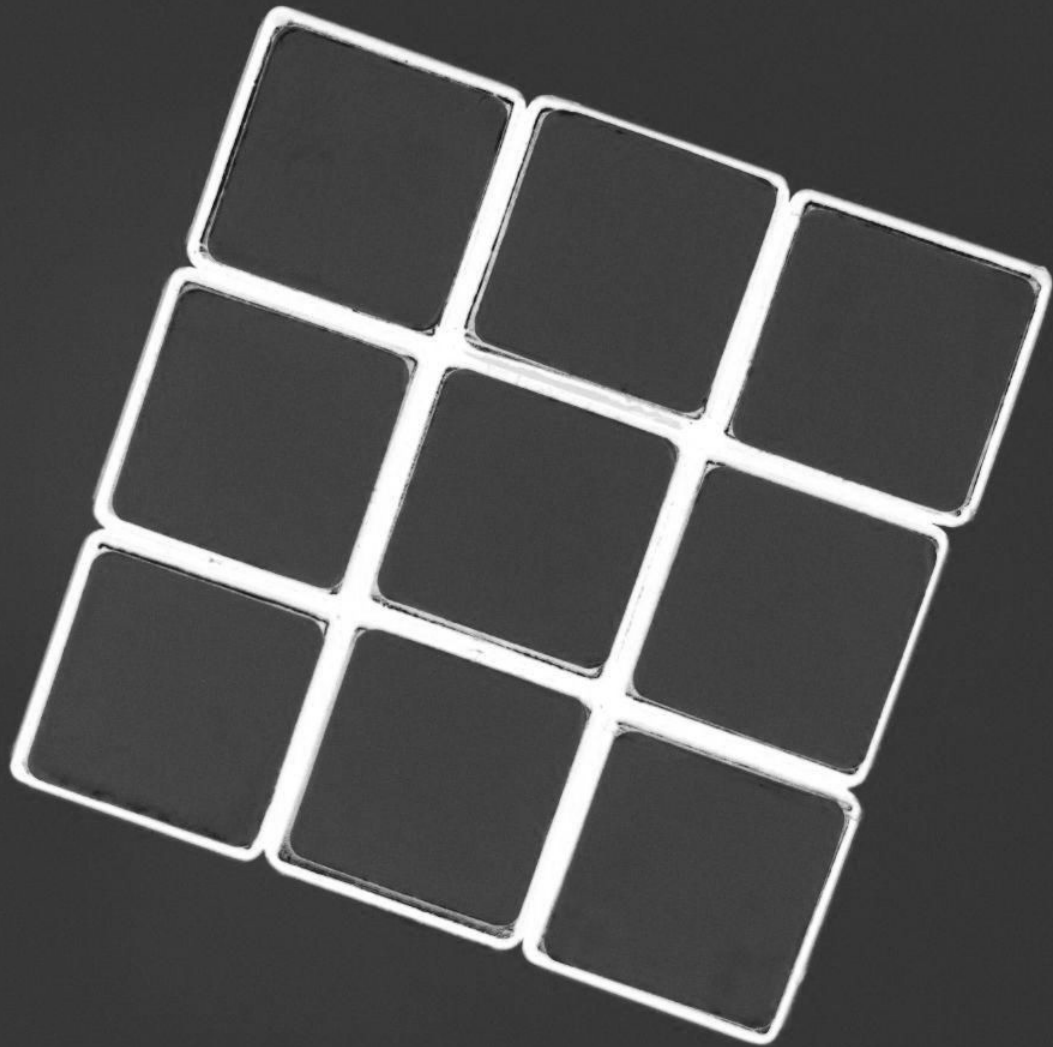
Desplegando aplicaciones con Open Application Model

Daniel Higuero
daniel@napptive.com



NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

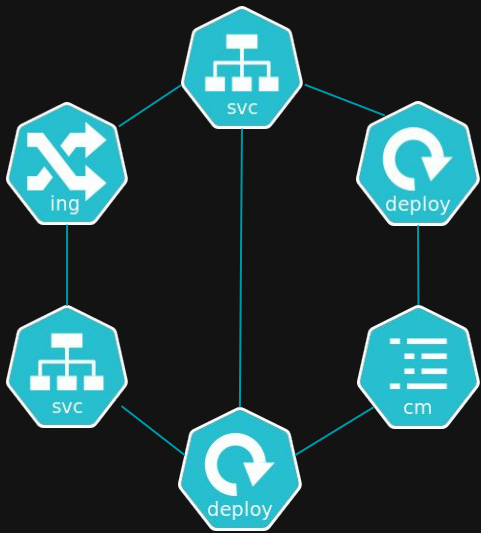
¿Que
problemas
aborda?





NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

Concepto de aplicación



No hay concepto de aplicación en kubernetes

Incremento de complejidad de las aplicaciones



NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

Fullⁿ Stack Developer



Perfiles complicados de
encontrar

No es fácil aprender
Kubernetes

Mi K8s != tu K8s



Muchas entidades dependen
del cloud

Estrategias multcloud y
posibilidad de migración

Cloud vs On-premise



NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

Open Application Model



Open Application Model (OAM)

- Lenguaje de especificación para aplicaciones
 - Agnóstico al proveedor/runtime
- Punto de vista de la aplicación
 - Qué necesito, no cómo lo obtengo
- Grandes proveedores de cloud involucrados



OAM

<https://oam.dev>



Microsoft



Alibaba Cloud

ORACLE®



Timeline





NAPTIVE
YOUR ONE-STOP CLOUD-NATIVE PLATFORM

Nuevas entidades



Entidades principales





Componente

- Se asocia con un μ -servicio
- Similar a un pod
- Capacidad de definir parámetros

```
apiVersion: core.oam.dev/v1alpha2
kind: Component
metadata:
  name: wordpress
spec:
  workload:
    apiVersion: core.oam.dev/v1alpha2
    kind: ContainerizedWorkload
    ...
    spec:
      containers:
        - name: wordpress
          image: bitnami/wordpress:5.7.0
          env:
            ...
parameters:
  - name: host
    required: true
    fieldPaths:
      - "spec.containers[0].env[0].value"
```




Traits

- Extiende las capacidades de un componente
- Implementado a través de operadores

```
apiVersion: core.napptive.com/v1alpha1
kind: Ingress
metadata:
  name: wp-ingress
spec:
  name: wordpress
  protocol: HTTP
  port: 8080
  path: /
  rewritePath: /
```



Application

- Definición de los componentes
- Parametrización
- Aplicación de traits

```
apiVersion: core.oam.dev/v1alpha2
kind: ApplicationConfiguration
metadata:
  name: my-wordpress
  ...
spec:
  components:
    - componentName: wordpress
      parameterValues:
        - name: dbpass
          value: "root"
        - name: host
          value: "mysql-db"
      traits:
        - trait:
            apiVersion: core.napptive.com/v1alpha1
            kind: Ingress
            metadata:
              name: wp-ingress
            spec:
              name: wordpress
              protocol: HTTP
              port: 8080
              path: /
              rewritePath: /
```



Scope

Similar a traits pero
aplicado a grupos de
componentes

- Estado de los servicios
- Seguridad
- Compartición
elementos

```
apiVersion: core.oam.dev/v1alpha2
kind: HealthScope
metadata:
  name: wordpress-health
spec:
  workloadRefs: []
---
apiVersion: core.oam.dev/v1alpha2
kind: ApplicationConfiguration
metadata:
  name: my-wordpress
spec:
  components:
    - componentName: wordpress
    ...
  scopes:
    - scopeRef:
      apiVersion: core.oam.dev/v1alpha2
      kind: HealthScope
      name: wordpress-health
```



NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

Demo
Time



talking



Wordpress & MySQL





NAPPTIVE
YOUR ONE-STOP CLOUD NATIVE PLATFORM

Thanks!

Any questions?

Let's talk

daniel@napptive.com

Estáis todos invitados!



playground.napptive.dev