



Utilizando Operator-Framework para automatizar aplicações no Kubernetes/OpenShift

CAMILA MACEDO

SR. Software Engineer

RED HAT CLOUD SERVICES

AGENDA

- INTRODUÇÃO
- ① CONTAINERS
- ② ORQUESTRAÇÃO DE CONTAINERS
- ③ CONFIGURAÇÕES NAS PLATAFORMAS
- K8S/OPENSIFT API'S
- RECURSOS CUSTOMIZADOS (CR/CRD)
- OPERATOR-FRAMEWORK
- DISPONIBILIZANDO O SEU OPERATOR
- POR ONDE COMEÇAR?
- DEMO
- PERGUNTAS ?

CONTAINERS

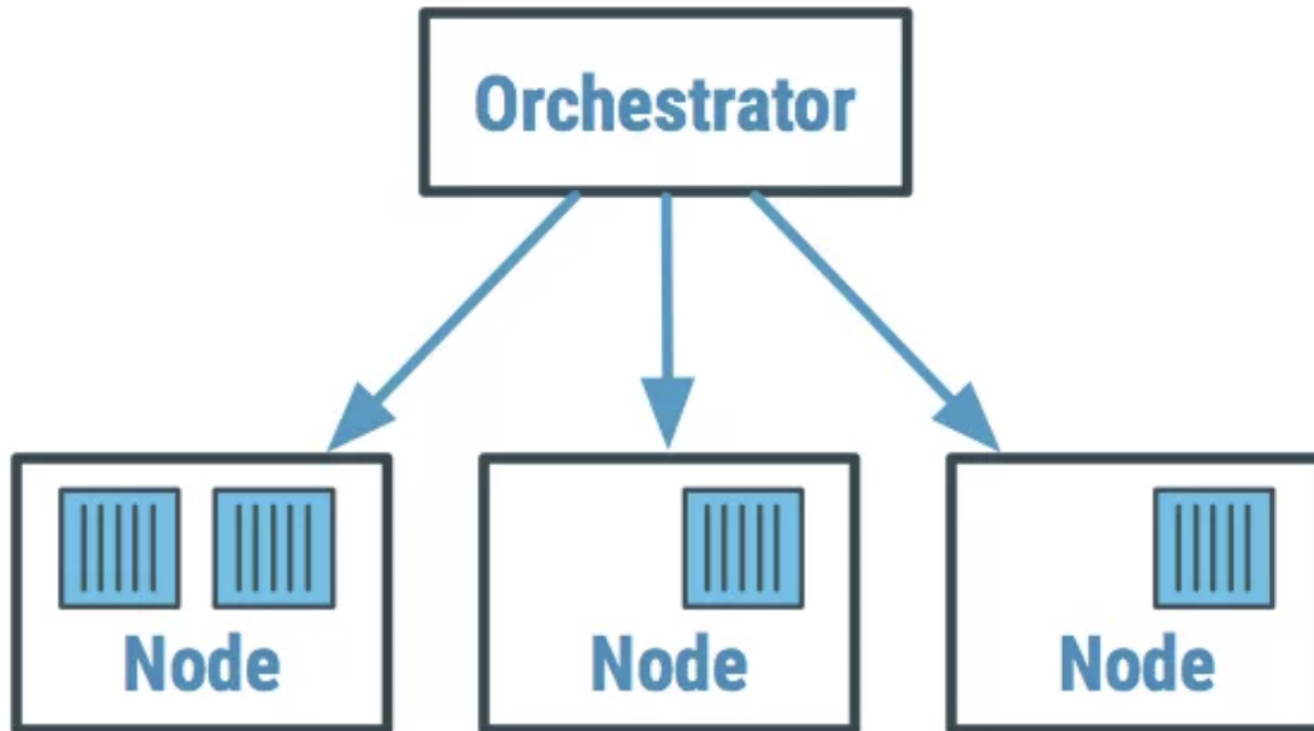
```
$ docker pull postgres
```

```
$ docker pull redis
```

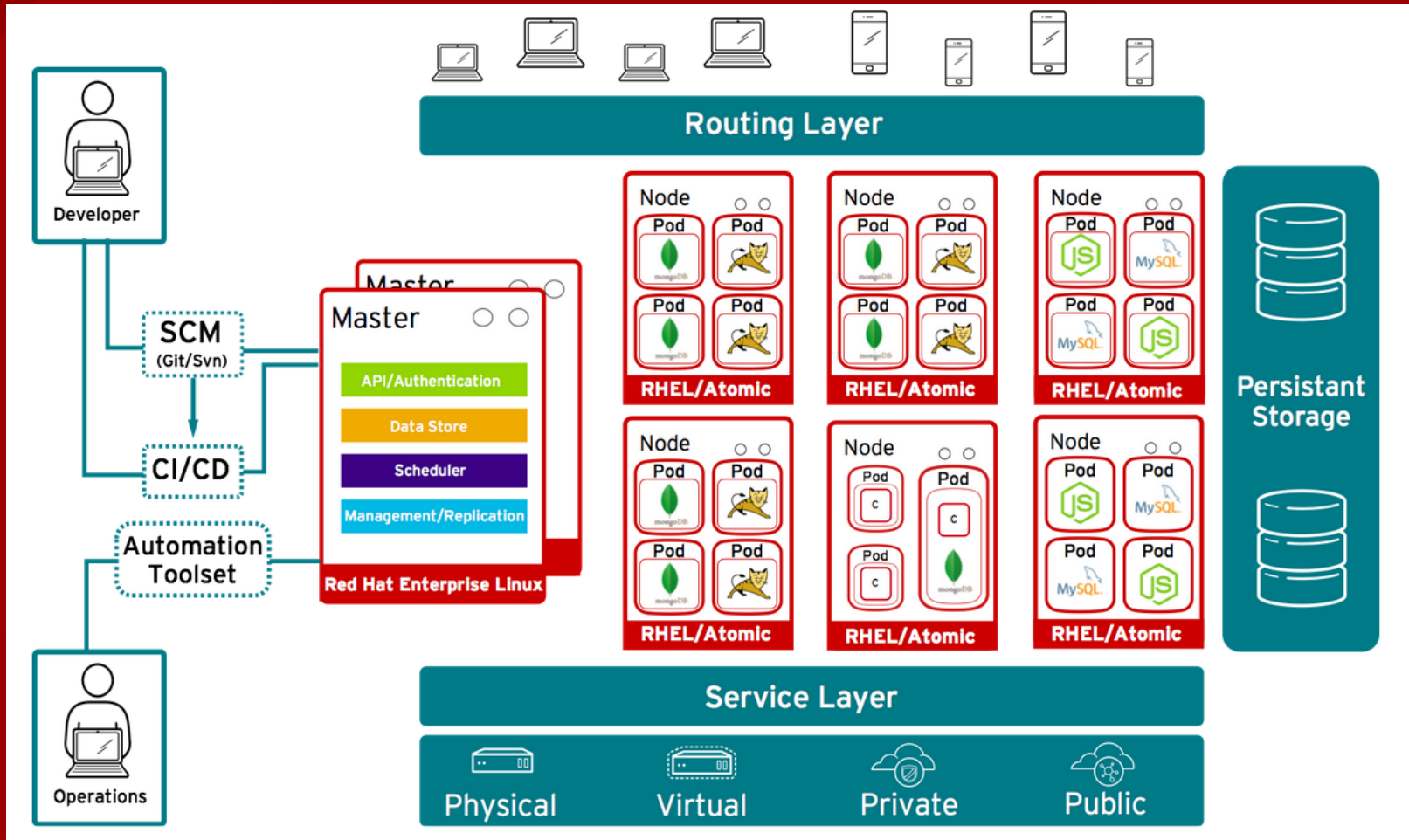
```
$ docker pull minhaaplicacao
```



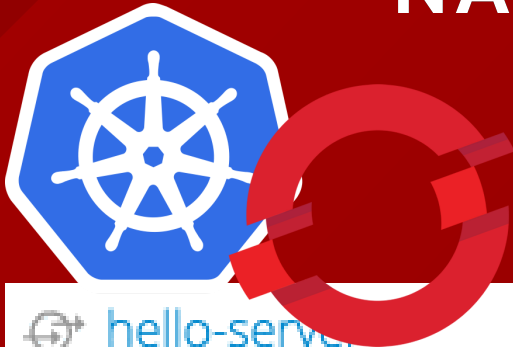
ORQUESTRAÇÃO DE CONTAINERS



ORQUESTRAÇÃO DE CONTAINERS



CONFIGURAÇÕES NAS PLATAFORMAS



 [hello-server](#)

Deployment Config [hello-server](#) – 5 days ago #1

CONTAINER: HELLO-SERVER

 **Image:** minishift-demo/hello-server

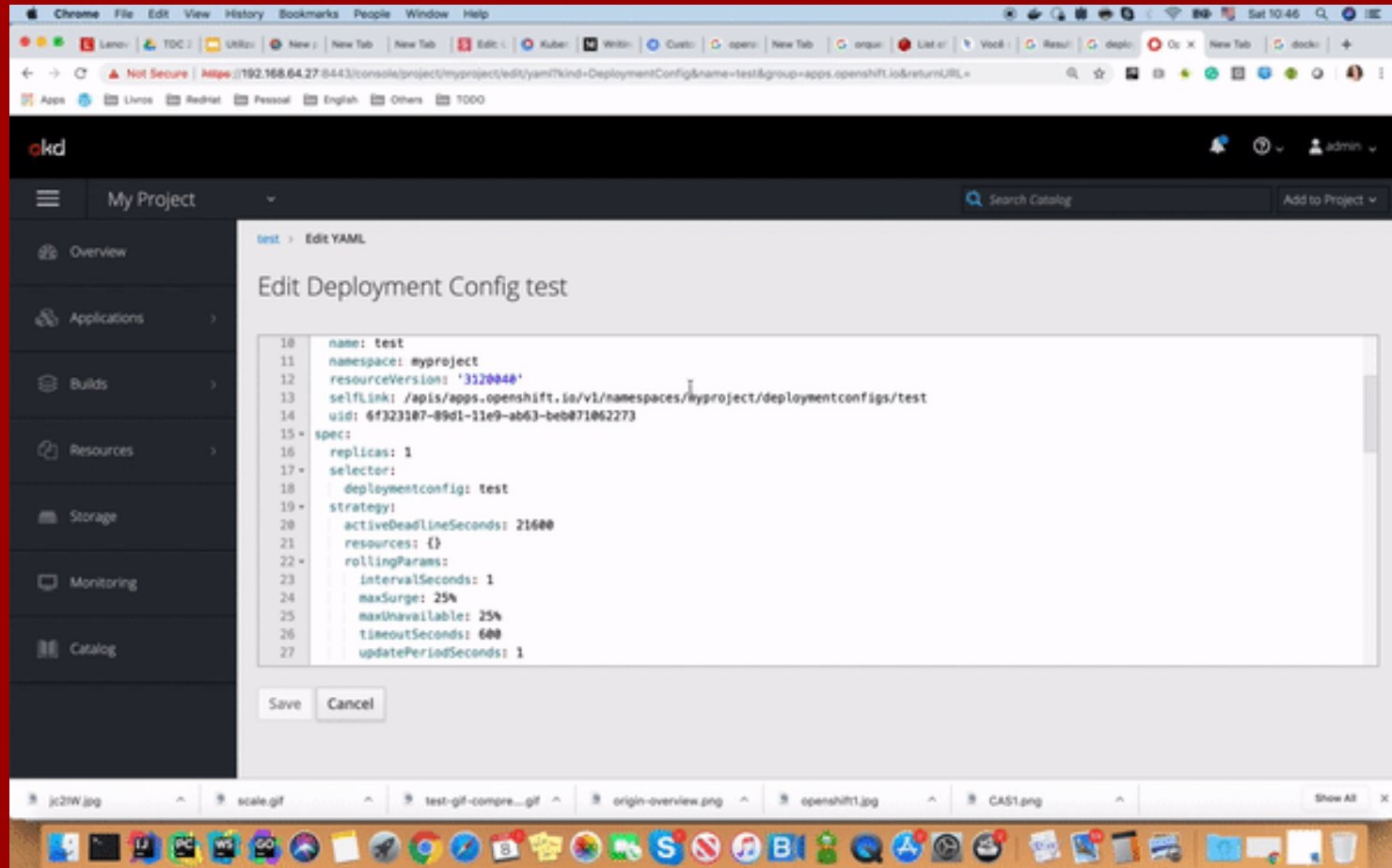
 **Ports:** 8080/TCP



1
pod




CONFIGURAÇÕES NAS PLATAFORMAS



K8S/OPENSHIFT API'S

```
1 # tells which API and version you're using to create this object
2 apiVersion: apps/v1
3 kind: Deployment # What kind of object you want to create
4 metadata: # Data that helps uniquely identify the object
5   name: redhat-app
6 spec:
7   # tells deployment to run 2 pods matching the template
8   replicas: 2
9   template:
10     metadata:
11       labels:
12         app: redhat-app
13     spec:
14       containers:
15         - name: redhat-app
16           # tells the docker image and tag which should be used
17           image: docker.io/cmacedo/redhat-app:master
18           ports:
19             - containerPort: 80
20
```


KS8/OPENSHIFT API'S

← → ↺ 🔒 https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.14/#deployment-v1-apps

Apps

W

Livros

RedHat

Pessoal

English

Others

TODO

Deployment v1 apps

Write Operations

Read Operations

Status Operations

Misc Operations

Job v1 batch

Pod v1 core

ReplicaSet v1 apps

ReplicationController v1 core

StatefulSet v1 apps

SERVICE APIS

Endpoints v1 core

Ingress v1beta1 networking.k8s.io

Service v1 core

CONFIG AND STORAGE APIS

ConfigMap v1 core

CSIDriver v1beta1 storage.k8s.io

CSINode v1beta1 storage.k8s.io

Secret v1 core

PersistentVolumeClaim v1 core

StorageClass v1 storage.k8s.io

Volume v1 core

VolumeAttachment v1 storage.k8s.io

METADATA APIS

ControllerRevision v1 apps

CustomResourceDefinition v1beta1 apiextensions.k8s.io

Event v1 core

LimitRange v1 core

HorizontalPodAutoscaler v1 autoscaling

MutatingWebhookConfiguration v1beta1 admissionregistration.k8s.io

ValidatingWebhookConfiguration v1beta1 admissionregistration.k8s.io

Deployment v1 apps

kubectl example

curl example

Group	Version	Kind
apps	v1	Deployment

Other API versions of this object exist: [v1beta2](#) [v1beta1](#) [v1beta1](#)

Appears In:

- DeploymentList [apps/v1]

Field	Description
apiVersion <i>string</i>	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value of the API and use the latest internal value of the schema. More info: https://git.k8s.io/community/contributors/devel/api-conventions.md#resources
kind <i>string</i>	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. More info: https://git.k8s.io/community/contributors/devel/api-conventions.md#types-kinds
metadata <i>ObjectMeta</i>	Standard object metadata.
spec <i>DeploymentSpec</i>	Specification of the desired behavior of the Deployment.
status <i>DeploymentStatus</i>	Most recently observed status of the Deployment.

DeploymentSpec v1 apps

Appears In:

- Deployment [apps/v1]

CRIANDO RECURSOS CUSTOMIZADOS

(CUSTOM RESOURCES/CUSTOM RESOURCES DEFINITIONS)

```
1 apiVersion: redhat.example.com/v1alpha1
2 kind: App
3 metadata:
4   name: redhat-app
5 spec:
6   nome: "Segredo do Universo"
7   id: 42
```

“ CR: Objeto que estende a API do Kubernetes ou permite que você introduza sua própria API em um projeto ou cluster.

“ CRD: Define seus próprios tipos de objeto e permite que o servidor de API manipule todo o ciclo de vida.

OPERATOR-FRAMEWORK

“ O Operator Framework é um projeto OpenSource o qual fornece ferramentas e recursos com o objetivo de auxiliar o desenvolvimento de projetos que realizam "operações" utilizando as API's em tempo de execução no cluster.



Operator Framework

The Operator Framework is an open source toolkit to manage Kubernetes native applications, called Operators, in an effective, automated, and scalable way.

<https://operatorhub.io/> Verified

[Report abuse](#)

Repositories 24

People 16

Pinned repositories

**operator-sdk**

SDK for building Kubernetes applications. Provides high level APIs, useful abstractions, and project scaffolding.

Go ★ 1.9k 🍴 445

**operator-lifecycle-manager**

A management framework for extending Kubernetes with Operators

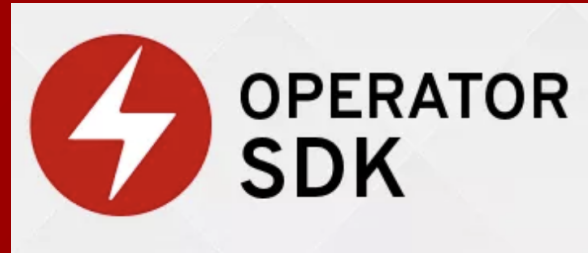
Go ★ 346 🍴 144

**operator-metering**

Operator metering is responsible for collecting metrics and other information about what's happening in a Kubernetes cluster, and providing a way to create reports on the collected data.

Go ★ 167 🍴 37

OPERATOR-FRAMEWORK

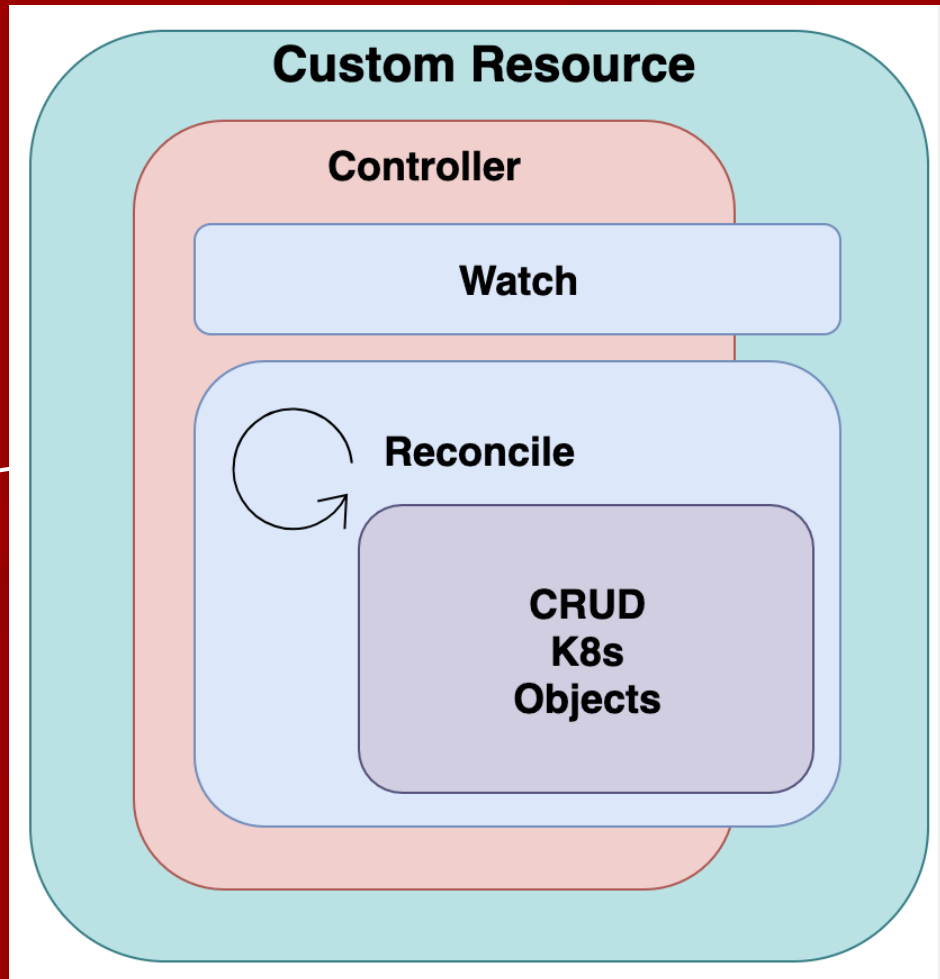
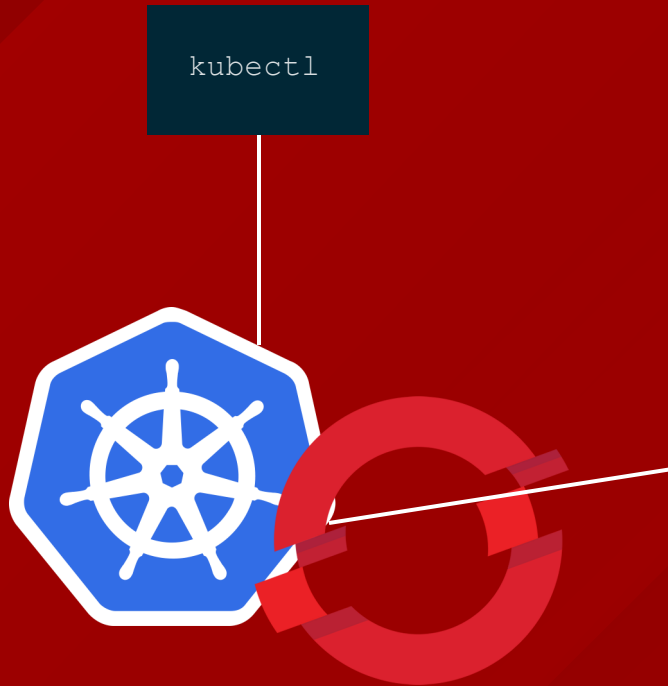


```
# Cria novo projeto
$ operator-sdk new redhat-operator
$ cd redhat-operator

# Adiciona no projeto AppService CR e CRD (API para AppService)
$ operator-sdk add api --api-version=redhat-operator.example.com/v1alpha1 --kind=App

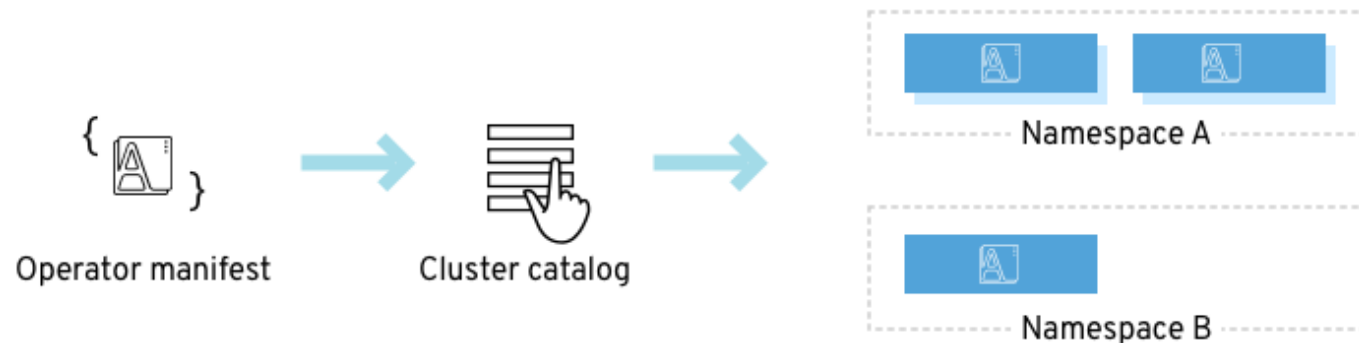
# Adiciona no projeto controller para AppService
$ operator-sdk add controller --api-version=redhat-operator.example.com/v1alpha1 --kind=App
```

OPERATOR-FRAMEWORK

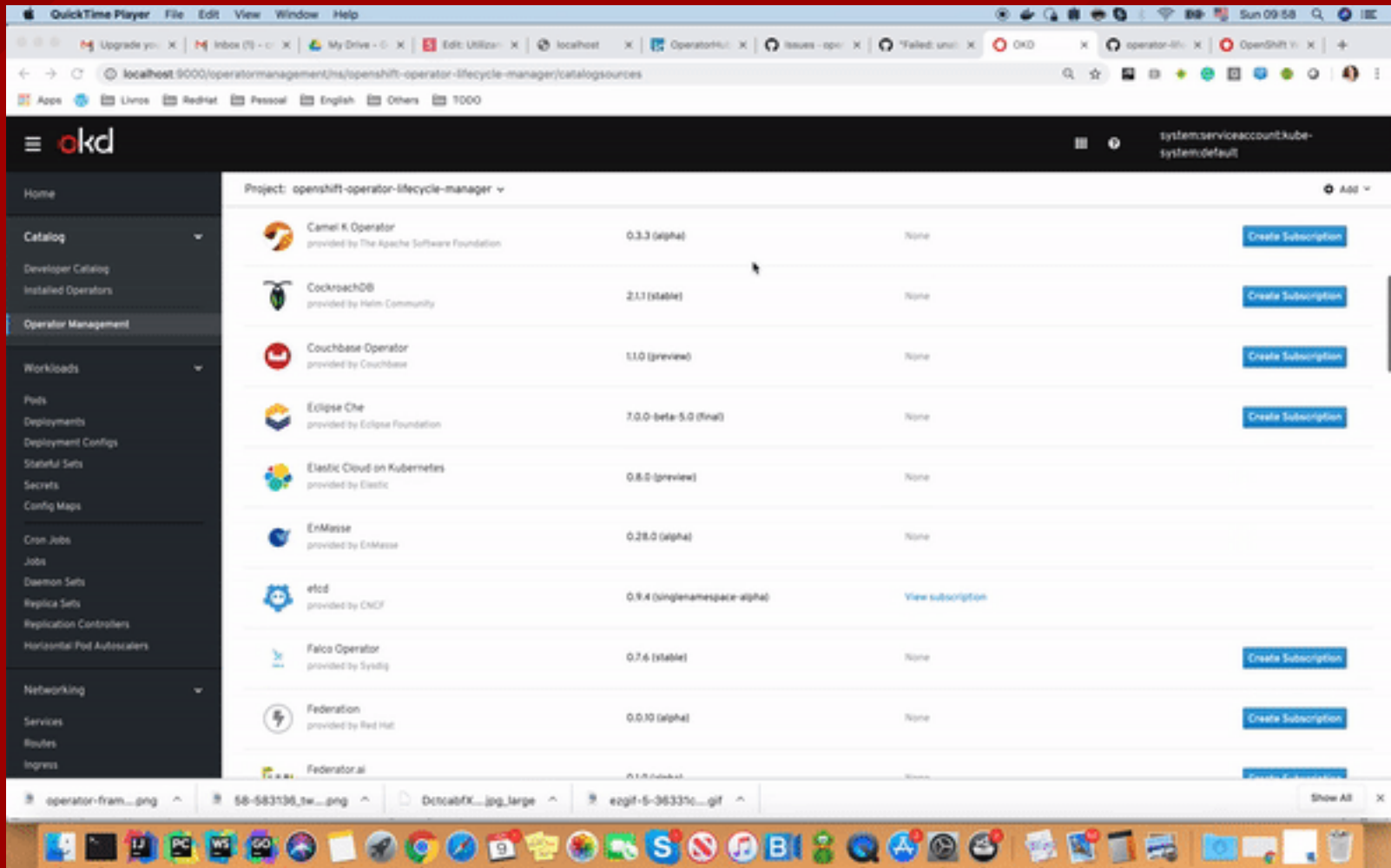


DISPONIBILIZANDO O SEU OPERATOR

Operator Lifecycle Manager *Install & update across clusters*



DISPONIBILIZANDO O SEU OPERATOR



The screenshot displays the OpenShift Operator Lifecycle Manager (OLM) catalog interface. The left sidebar shows the navigation menu with 'Operator Management' selected. The main content area lists several operators available for installation, each with a logo, name, provider, version, and a 'Create Subscription' button.

Operator	Version	Subscription
Camel K Operator provided by The Apache Software Foundation	0.3.3 (alpha)	None
CockroachDB provided by Helm Community	2.1.1 (stable)	None
Couchbase Operator provided by Couchbase	1.1.0 (preview)	None
Eclipse Che provided by Eclipse Foundation	7.0.0-beta-5.0 (final)	None
Elastic Cloud on Kubernetes provided by Elastic	0.8.0 (preview)	None
EnMasse provided by EnMasse	0.28.0 (alpha)	None
etcd provided by CNCF	0.9.4 (singlenamespace-alpha)	View subscription
Falco Operator provided by Sysdig	0.7.6 (stable)	None
Federation provided by Red Hat	0.0.10 (alpha)	None
Federator.ai	0.1.0 (alpha)	None

DISPONIBILIZANDO O SEU OPERATOR

OperatorHub.io

Search OperatorHub...[Contribute](#)

Welcome to OperatorHub.io

OperatorHub.io is a new home for the Kubernetes community to share Operators. Find an existing Operator or list your own today.

CATEGORIES

46 ITEMS

VIEW SORT A-Z

AI/Machine Learning

Big Data

Cloud Provider

Database

Developer Tools

Integration & Delivery

Logging & Tracing

Monitoring

Networking

OpenShift Optional

Security

Storage

Streaming & Messaging

Other

PROVIDER

☐ Amazon Web Services (1)

☐ Aqua Security (1)

☐ Banzai Cloud (2)

☐ CNCF (2)

**Aqua Security Operator**
provided by Aqua Security, Inc.

The Aqua Security Operator runs within Kubernetes cluster and provides a means to

**Couchbase Operator**
provided by Couchbase

The Couchbase Autonomous Operator allows users to easily deploy, manage, and maintain

**AWS Service Operator**
provided by Amazon Web Services, Inc.

The AWS Service Operator allows you to manage AWS

**Crunchy PostgreSQL Enterprise**
provided by Crunchy Data

PostgreSQL is a powerful, open source object-relational

**Camel K Operator**
provided by The Apache Software Foundation

Apache Camel K (a.k.a. Kamel) is a lightweight integration

**Dynatrace OneAgent**
provided by Dynatrace LLC

Install full-stack monitoring of Kubernetes clusters with the Dynatrace OneAgent.

**CockroachDB**
provided by Helm Community

CockroachDB Operator based on the CockroachDB helm chart

**Eclipse Che**
provided by Eclipse Foundation

A Kube-native development solution that delivers portable

**Community Jaeger Operator**
provided by CNCF

Provides tracing, monitoring and troubleshooting microservices-based

**Elastic Cloud on Kubernetes**
provided by Elastic

Run Elasticsearch and Kibana on Kubernetes

POR ONDE COMEÇAR?

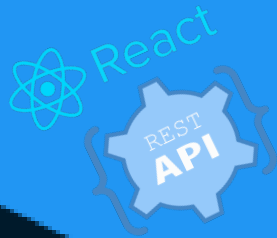
- **Getting-Started:** <https://github.com/operator-framework/getting-started/blob/master/README.md>
- **Blog:** <https://dev4devs.com/2019/03/08/kubernetes-openshift-operators-getting-started-from-a-to-z/>



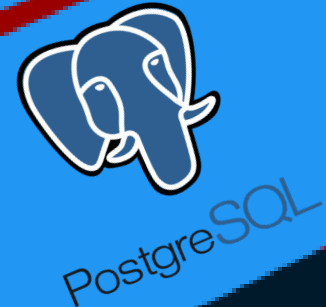
DEMO

MOBILE SECURITY SERVICE OPERATOR

Application (REST + UI)



DATABASE



RED HAT®
OPENSIFT
Container Platform

<https://github.com/aerogear/mobile-security-service-operator>

PERGUNTAS ?



Red Hat

MUITO OBRIGADA :-)

**EU ESPERO QUE VOCÊ
TENHA GOSTADO**