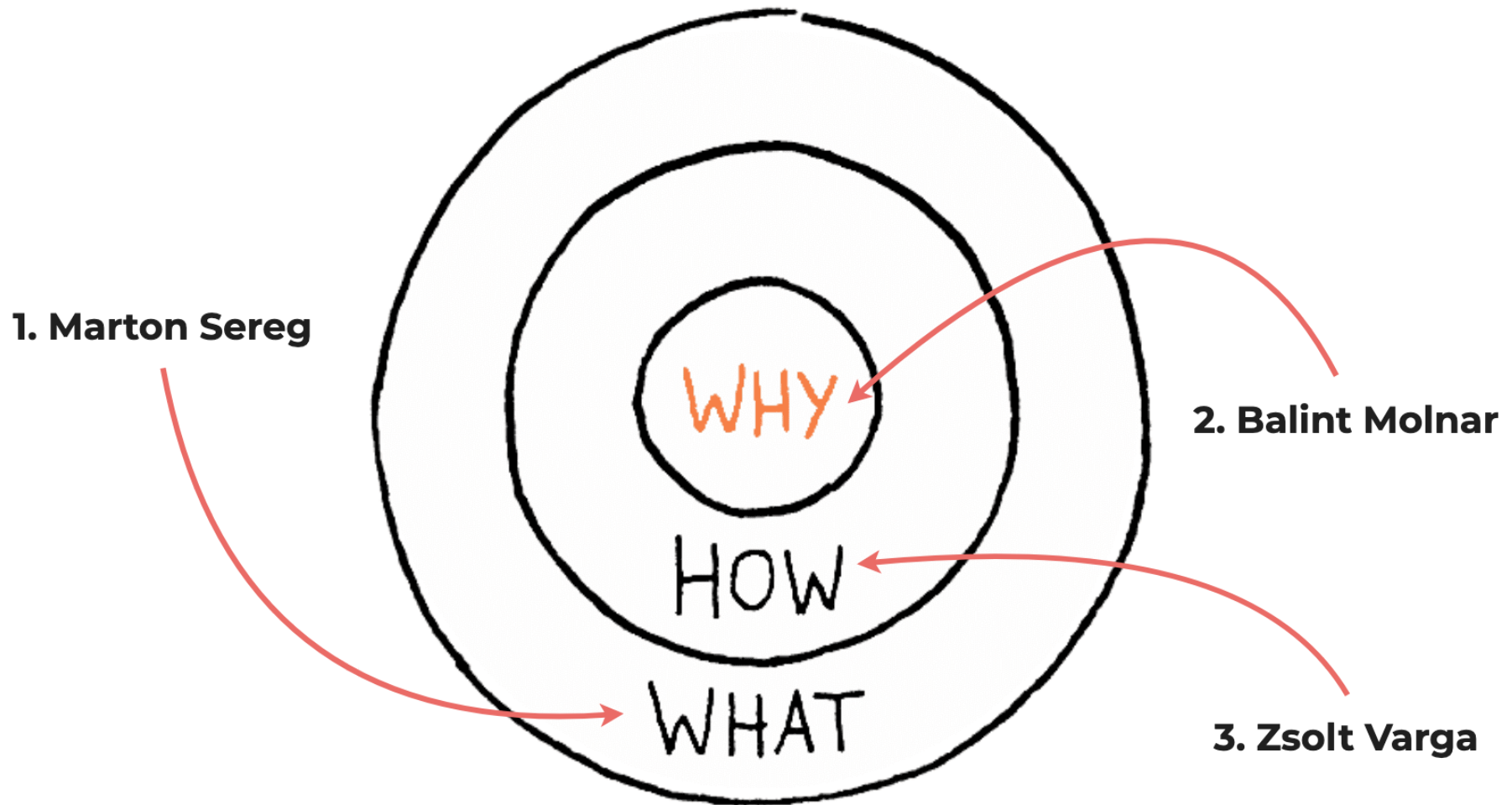


Kubernetes Operators - The basics

4 June 2019

Marton Sereg
Banzai Cloud

Agenda



Definition

Kubernetes controllers that implement an API for a specific application, such as Etcd, Prometheus or Vault are often referred to as Operators.

3

What is an API resource?

A Kubernetes Resource is a declarative API with a well defined Schema structure and endpoints.

Declarative API

A declarative API expresses a fixed state that the cluster must continually work towards.

Resource schema

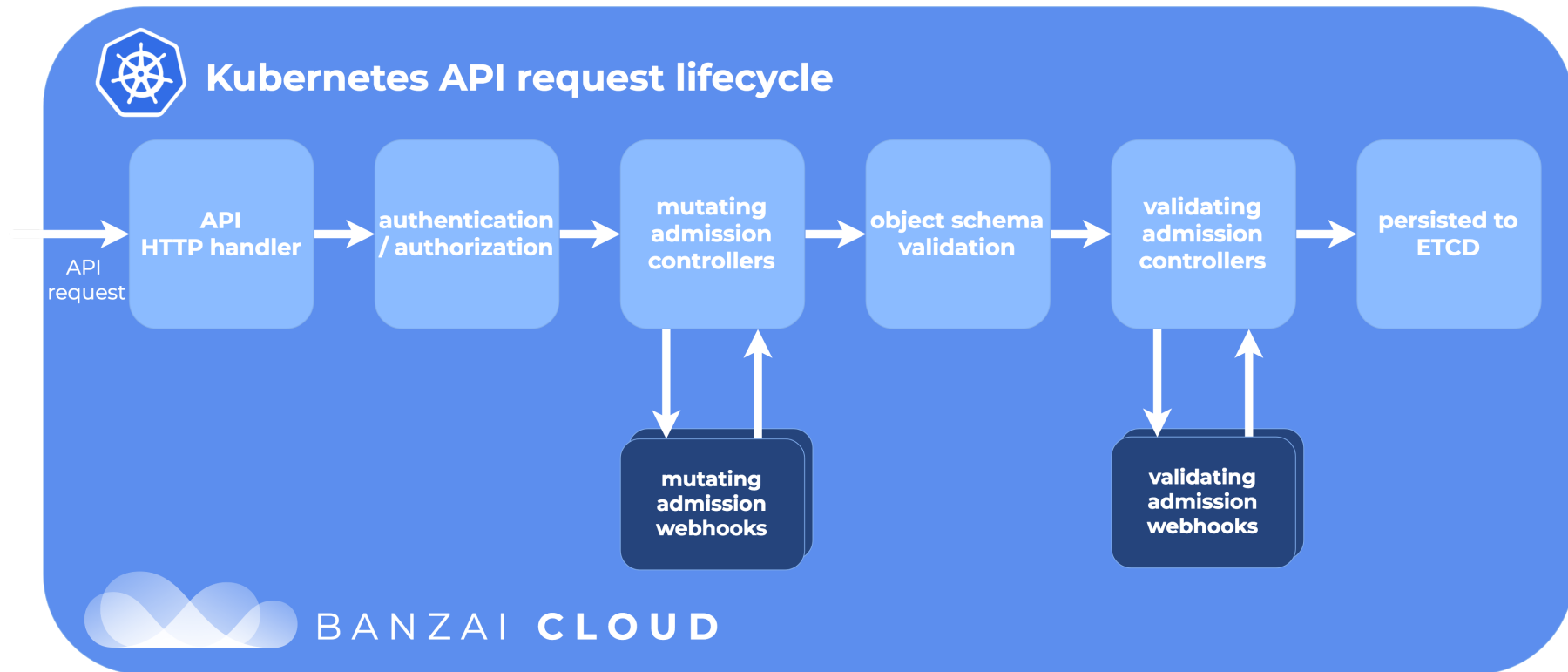
Every Kubernetes resource has a Group, Version and Kind that uniquely identifies it.

4

How does it look like?

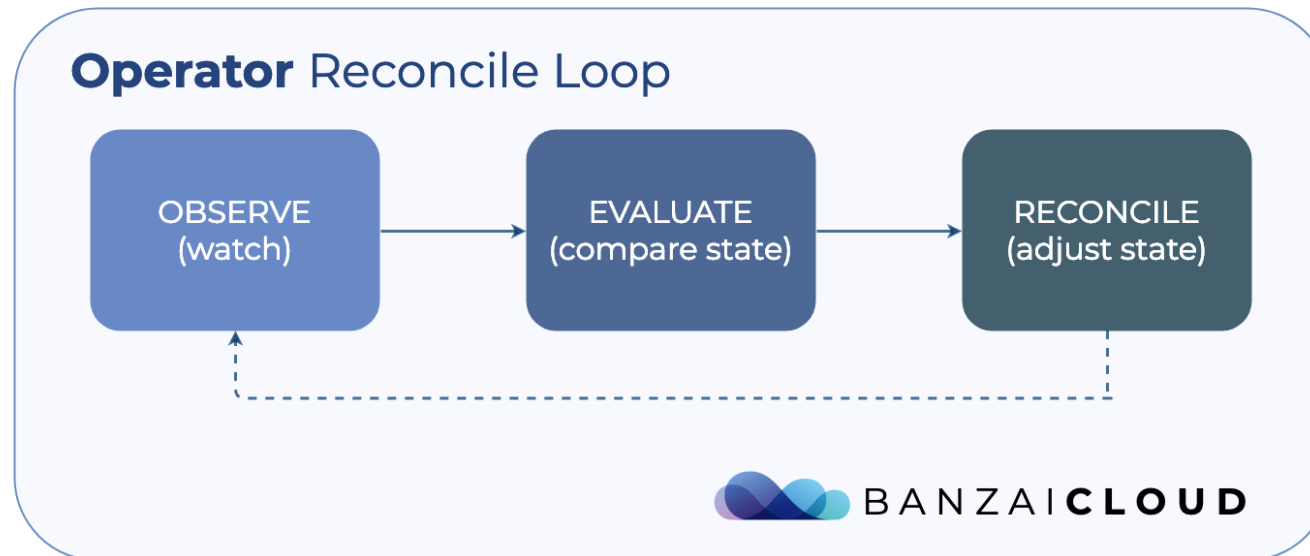
```
$ kubectl get deployment -n default nginx-deployment
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: nginx-deployment
  namespace: default
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - image: nginx:1.12.1
        imagePullPolicy: IfNotPresent
        name: nginx
status:
  availableReplicas: 2
  readyReplicas: 2
  replicas: 2
```

What happens when I create a resource?



What is a controller?

Controllers implement APIs defined by Resources. A Kubernetes Controller is a routine running in a Kubernetes cluster that **watches** for create / update / delete events on Resources, and triggers a **reconcile** function in response. Reconcile is a function that may be called at any time with the Namespace and Name of an object, and it will make the cluster state match the state declared in the object Spec.



Okay, but what does it have to do with operators?

```
$ kubectl get crd istios.istio.banzaicloud.io -o yaml
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
  name: istios.istio.banzaicloud.io
spec:
  group: istio.banzaicloud.io
  names:
    kind: Istio
    listKind: IstioList
    plural: istios
    singular: istio
  scope: Namespaced
  validation:
    openAPIV3Schema:
      [...]
  version: v1beta1
status:
  acceptedNames:
    kind: Istio
    listKind: IstioList
    plural: istios
    singular: istio
```


Custom resources are the new answer to the Ultimate Question of Life, the Universe, and Everything

```
$ kubectl get istios.istio.banzaicloud.io istio-sample -n istio-system -o yaml
apiVersion: istio.banzaicloud.io/v1beta1
kind: Istio
metadata:
  labels:
    controller-tools.k8s.io: "1.0"
  name: istio-sample
  namespace: istio-system
  [...]
spec:
  version: 1.1.7
  mtls: false
  outboundTrafficPolicy:
    mode: ALLOW_ANY
  [...]
status:
  ErrorMessage: ""
  GatewayAddress:
  - 35.197.211.202
  Status: Available
```

...for a specific application

- Prometheus (github.com/coreos/prometheus-operator (<https://github.com/coreos/prometheus-operator>))
- Etcd (github.com/coreos/etcd-operator (<https://github.com/coreos/etcd-operator>))
- Vault (github.com/banzaicloud/bank-vaults (<https://github.com/banzaicloud/bank-vaults>))
- Istio (github.com/banzaicloud/istio-operator (<https://github.com/banzaicloud/istio-operator>))
- Kafka (github.com/banzaicloud/kafka-operator (<https://github.com/banzaicloud/kafka-operator>))
- Fluentd / Fluentbit (github.com/banzaicloud/logging-operator (<https://github.com/banzaicloud/logging-operator>))
- etc... (see operatorhub.io (<https://operatorhub.io>))

How do I create a new resource or a controller?

1. Write it from scratch
2. Use Kubebuilder
3. Use Operator SDK

Summary

An Operator is a method of packaging, deploying and managing a Kubernetes application.

Operational knowledge of the application is baked in - it helps with automation.

Smarter and more tailored than generic tools - but needs coding.

Uses Kubernetes constructs such as CRDs, controllers and the reconciliation loop.

12

Thank you

Marton Sereg

Banzai Cloud

marton@banzaicloud.com (mailto:marton@banzaicloud.com)

<https://github.com/martonsereg/k8s-meetup> (https://github.com/martonsereg/k8s-meetup)