



What is Kubernetes?

What is Kubernetes?

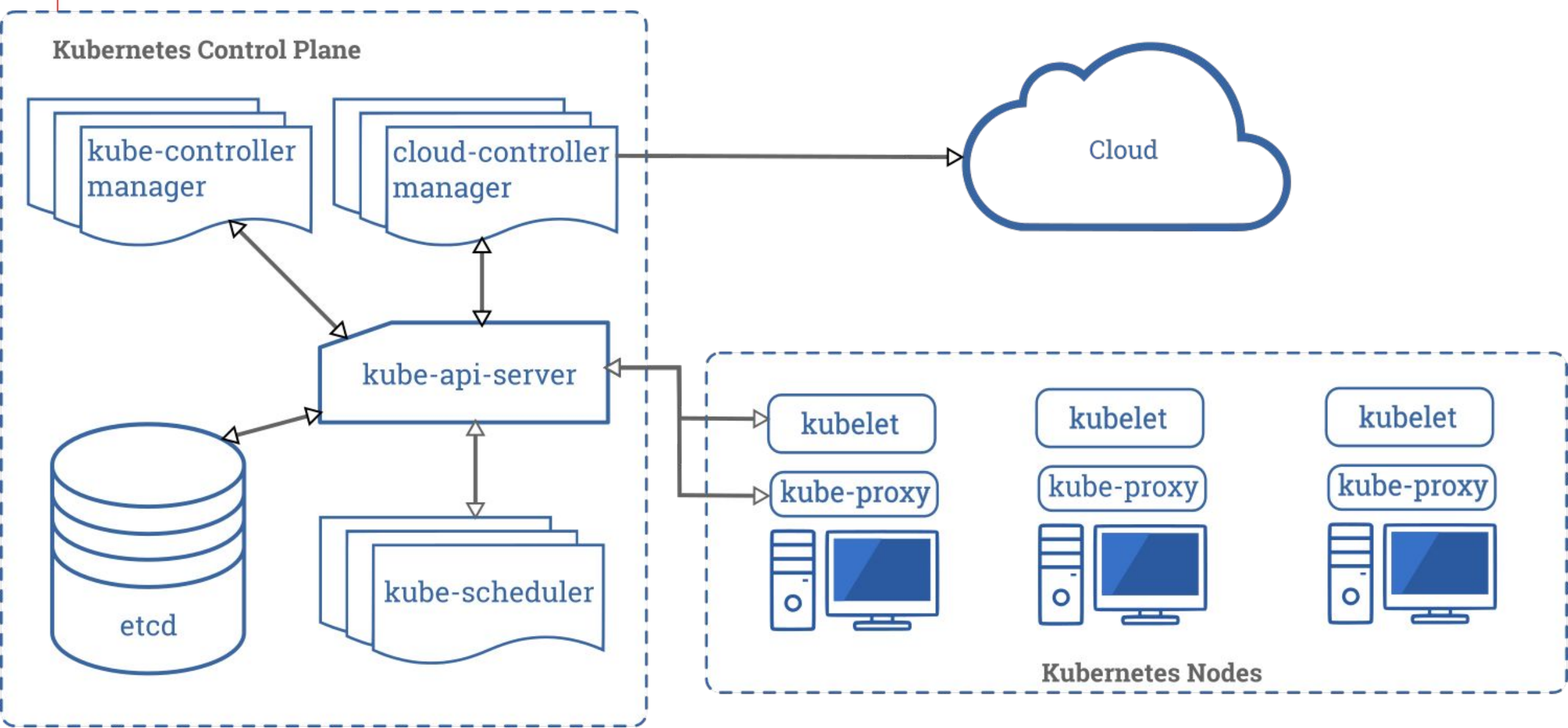
- “An open-source system for automating deployment, scaling, and management of containerized applications” (<https://kubernetes.io/>)
- Layman terms – Kubernetes is a scheduler for containers
- Abstracts away the details of infrastructure
- “Kubernetes is the new kernel.”



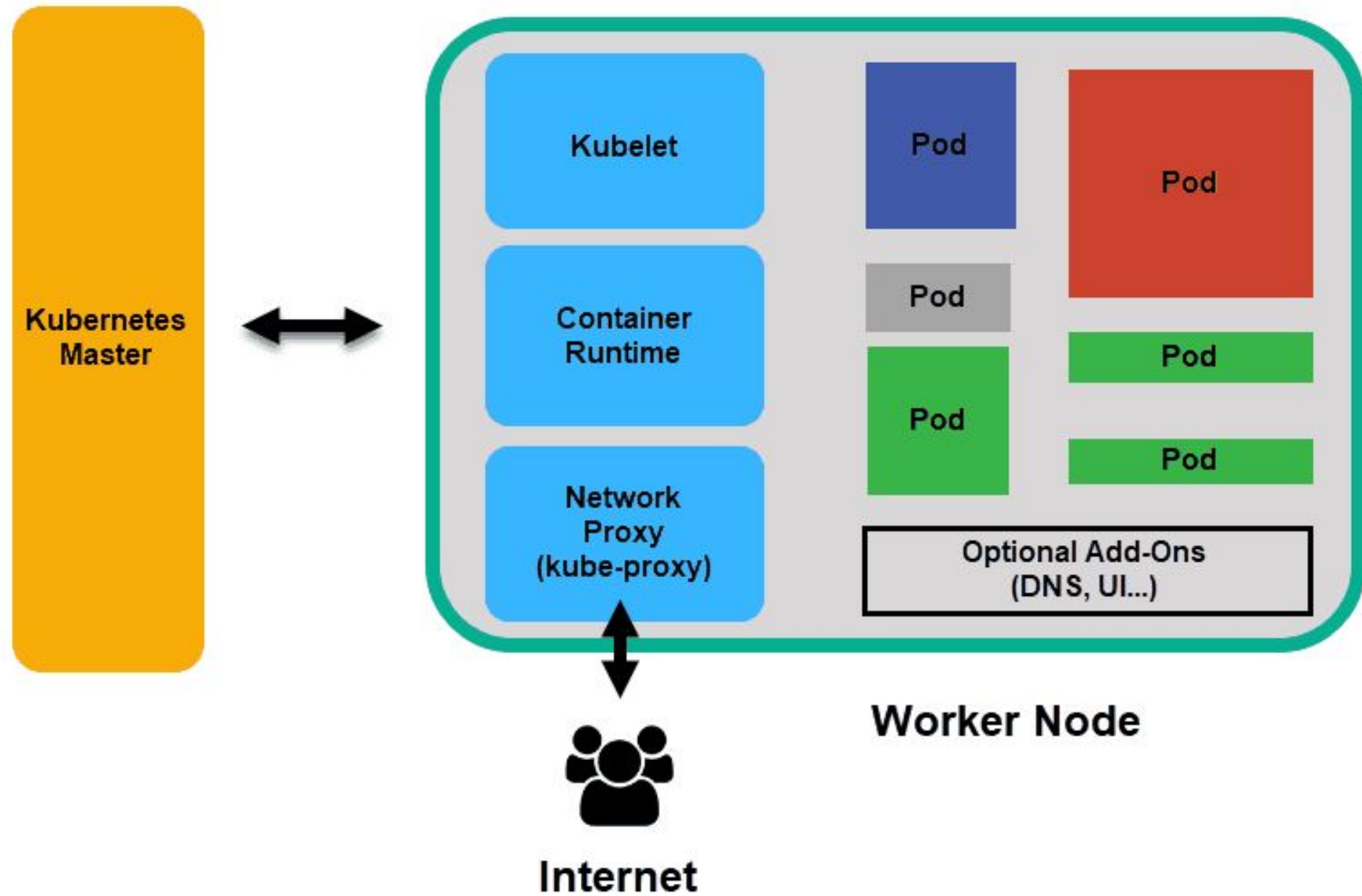
History of Kubernetes

- Came from Google (announced in 2014)
- Influenced by Google's own cluster manager (Borg/Omega)
- Greek word for Helmsman or Governor
 - The captain of the container ship
- Written in GOLANG
- Google and the Linux foundation formed the CNCF (Cloud Native Computing Foundation) and donated the first project (2015)





What is





Kubernetes basic building blocks

Declarative API – The unsung hero

The Control Plane's Declarative API is the most underrated and important part of Kubernetes

- Inception: Orchestrator of containers
- Today: Orchestrator of containers and container adjacent things
- Tomorrow: Orchestrator of Orchestrators and Clouds

"Kubernetes is Infrastructure as Data" – Kelsey Hightower

"Containers is just how it started, Kubernetes is bound for much more" – Bassam Tabbara

Object Model

All Objects have:

- GVK (Group/Version/Kind)
- Metadata
- "spec" - This is what I want to happen
- "status" - This is what the actual state is

**Objects are described in YAML and converted to JSON when sent to the API server.*

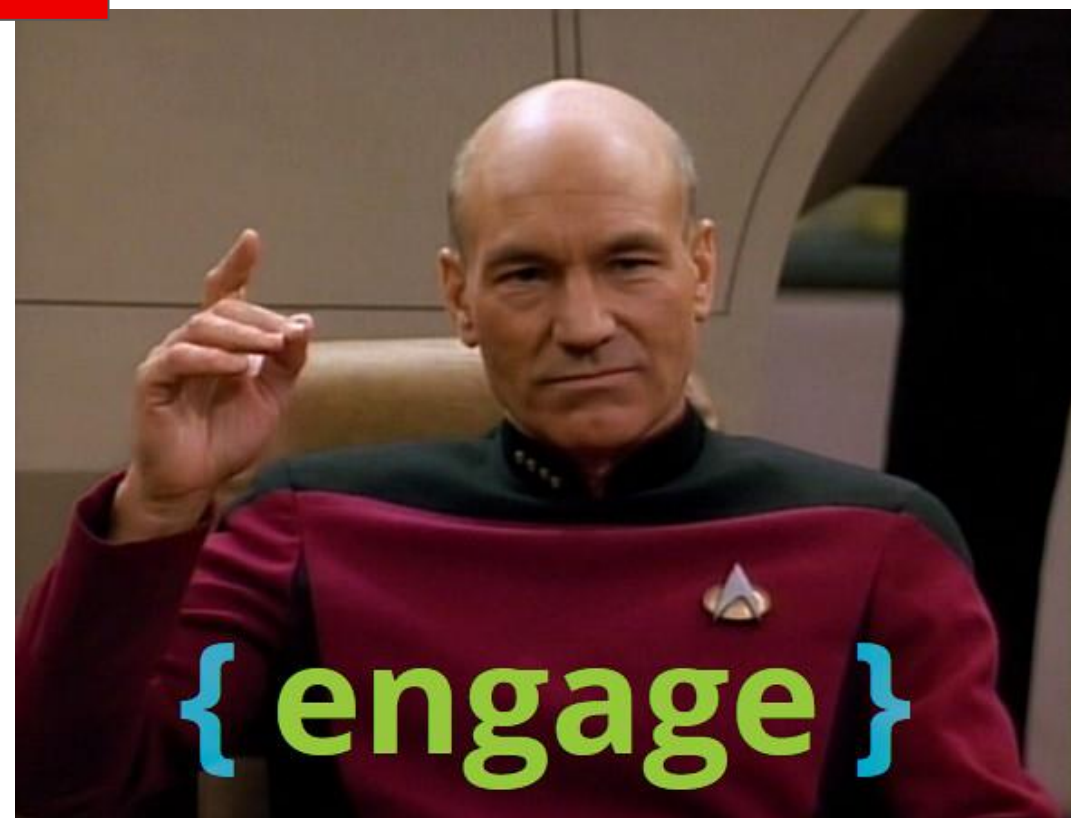
Resource definition

```
object1.yaml x
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: hello-world
        image: hello-world:latest
        ports:
        - containerPort: 80
```

Group: apps Version: v1beta

Kind: Deployment

Spec: Make it so!

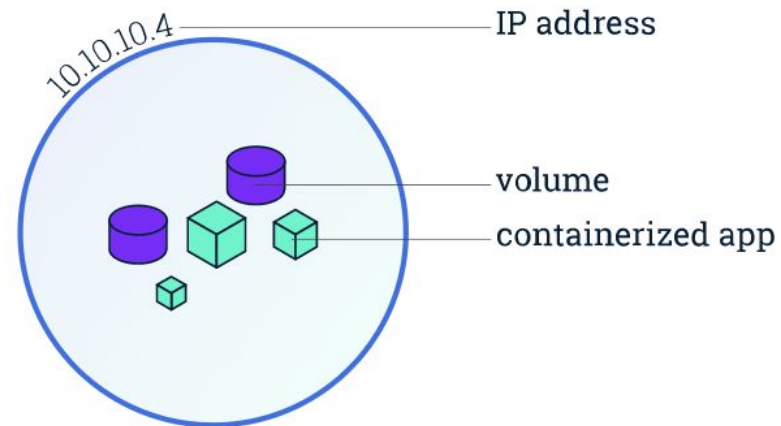


Pod

A pod is the atomic unit of an application in Kubernetes.

One (or more) containers that share:

- Networking (IP address)
- Linux namespace
- Storage
- Memory



Pod

```
kubectl create -f  
https://raw.githubusercontent.com/jankleinert/hello-workshop/master/pod.json
```

```
kubectl get pods
```

```
kubectl describe pod/hello-k8s
```

Volumes / Configmaps / Secrets

Volumes, Configmaps, and Secrets are used to insert data into pods at runtime

- **Volumes** are essentially just a directory on disk
 - Ephemeral or Persistent (PV/PVC)
- **Configmaps** are a declarative way to store and insert configuration data in to pods as volumes
- **Secrets** are just like configmaps only they store their data in RAM and can be encrypted and obfuscated from the host.

Service

A networking construct to abstract the Pod

3 basic types of Services:

- ClusterIP – basic load balancing (internal to cluster only)
- NodePort – Translates ClusterIP:port to NodeIP:port
- LoadBalancer – integrates with 3rd party / external LB



Service

Acts as a single endpoint for a collection of replicated pods like a load balancer



```
kind: Service
apiVersion: v1
metadata:
  name: hello-k8s
  creationTimestamp:
  labels:
    run: hello-k8s
spec:
  ports:
    - protocol: TCP
      port: 8080
      targetPort: 8080
  selector:
    run: hello-k8s
  type: NodePort
status:
  loadBalancer: {}
```

Service

```
kubectl expose pod/hello-k8s --port 8080 --type=NodePort
```

```
kubectl get svc/hello-k8s -o yaml
```

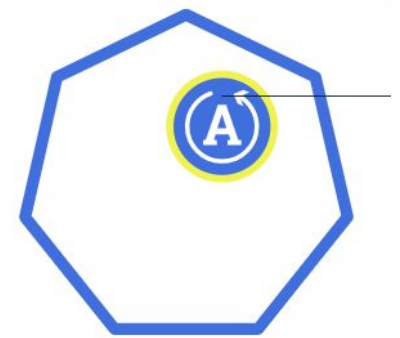
```
curl hello-k8s.<userX>:8080
```

ReplicaSets / Deployments

ReplicaSets - Scales Pods...That's it...

Deployments - Declarative updater for pods and ReplicaSets

- What is running and where?
- What services and resources are available to it?
- Policies for how things should behave (IFTTT logic)
- Rollback / Scaleup / Rollout definitions



Stateful Sets / Daemon Sets / Cron Jobs

Stateful Sets

- A special deployment type for stateful workloads
- Stable and ordered constructs
- Volumes scale out with pods

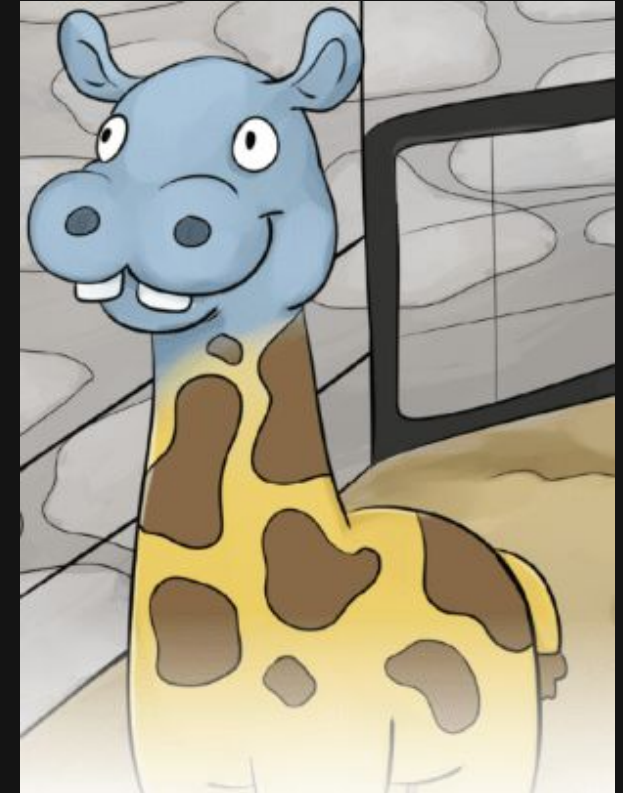
Daemon Sets

- A special deployment type that runs on a defined set of nodes
- As nodes get added that match, the pods scale
- Useful for infrastructure services

Cron Jobs

- Batch processing
- Serverless (like)
- Schedule **when** you want to run a pod

Extending Kubernetes: Custom Resource Definitions



Custom Resource Definitions

- Extend the Kubernetes API with your own GVK
- Inherit the rich high-level API abstractions and data model
- Your resources become first class citizens in the Kubernetes ecosystem
- Operator pattern pairs custom resources with custom controllers to program specific knowledge of your applications into the Kubernetes ecosystem.

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)

[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



twitter.com/RedHat