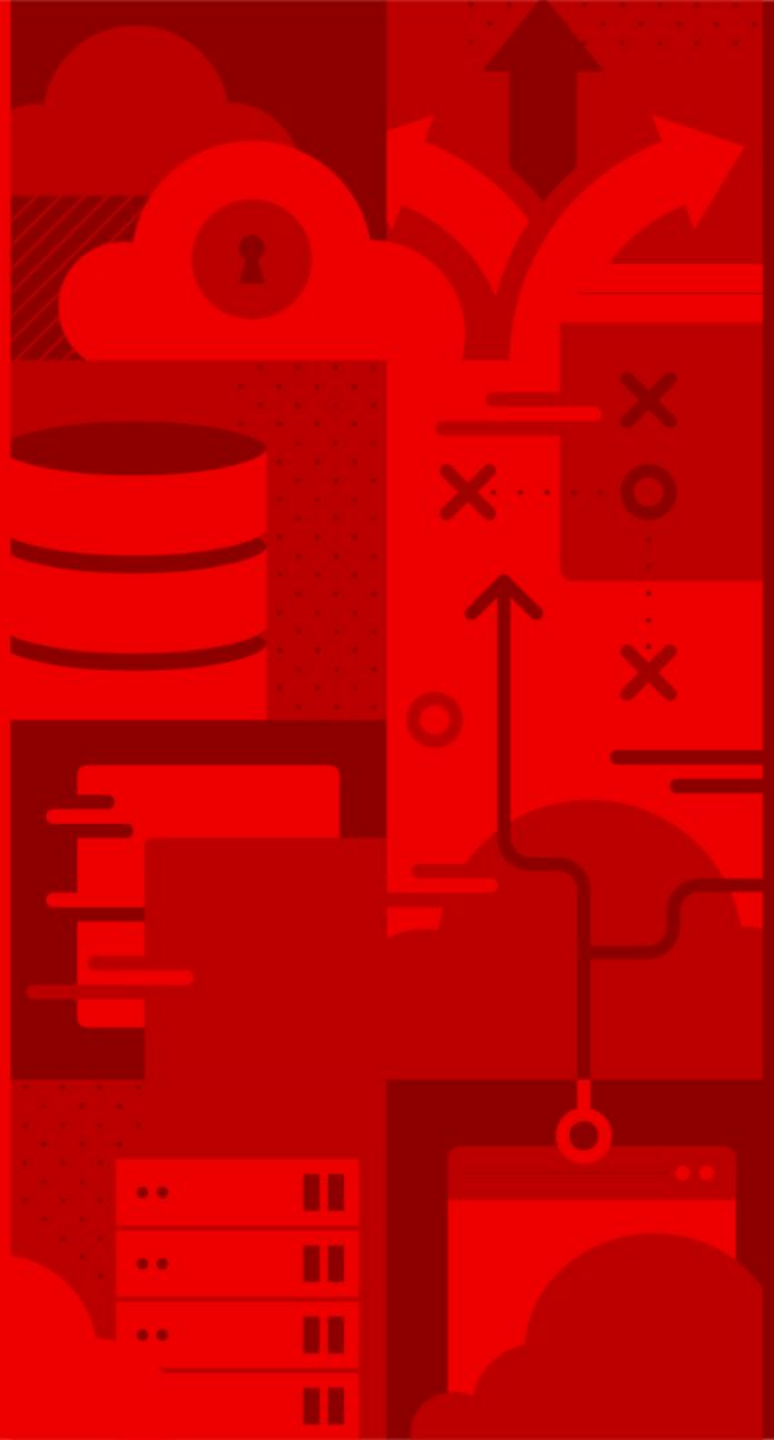


What is Kubernetes?

Red Hat

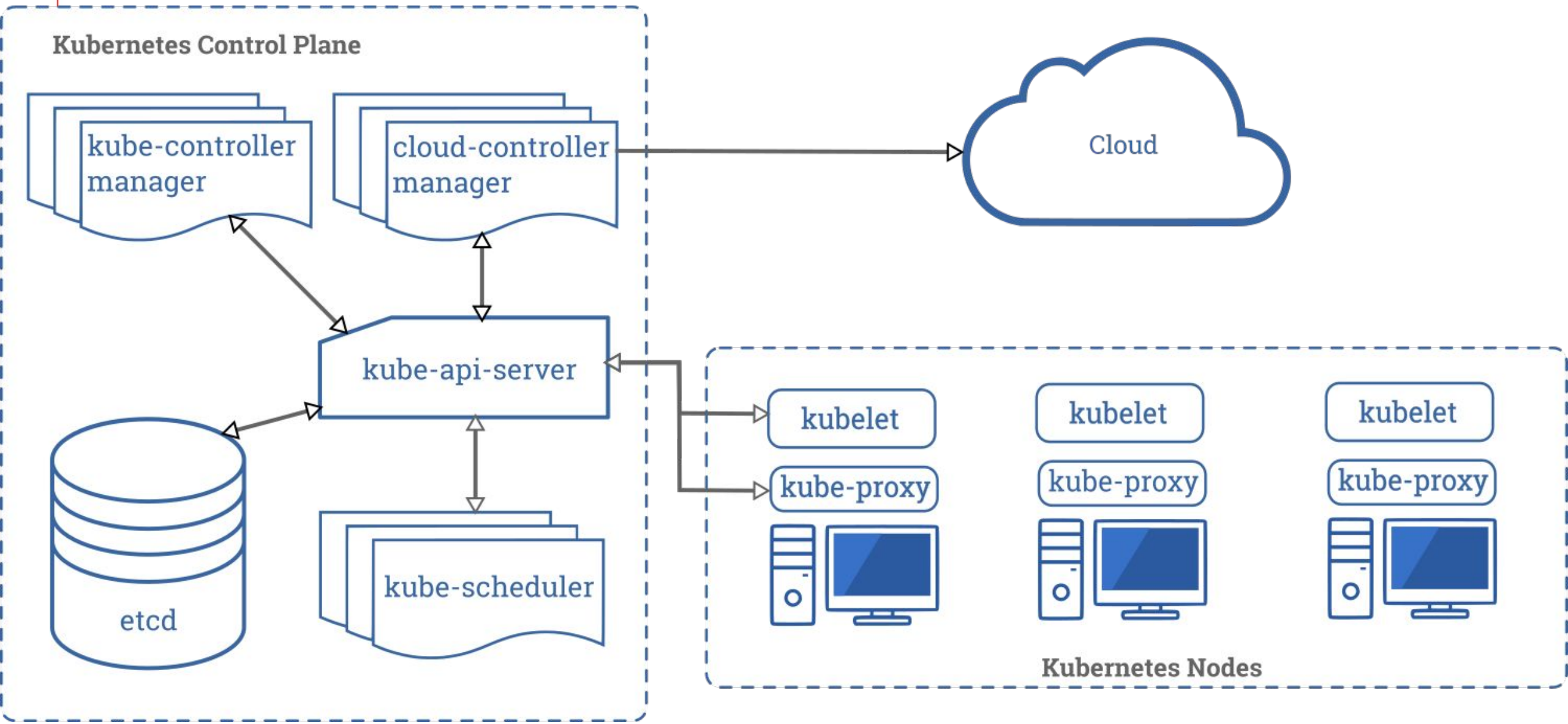
What is Kubernetes?

- “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.”

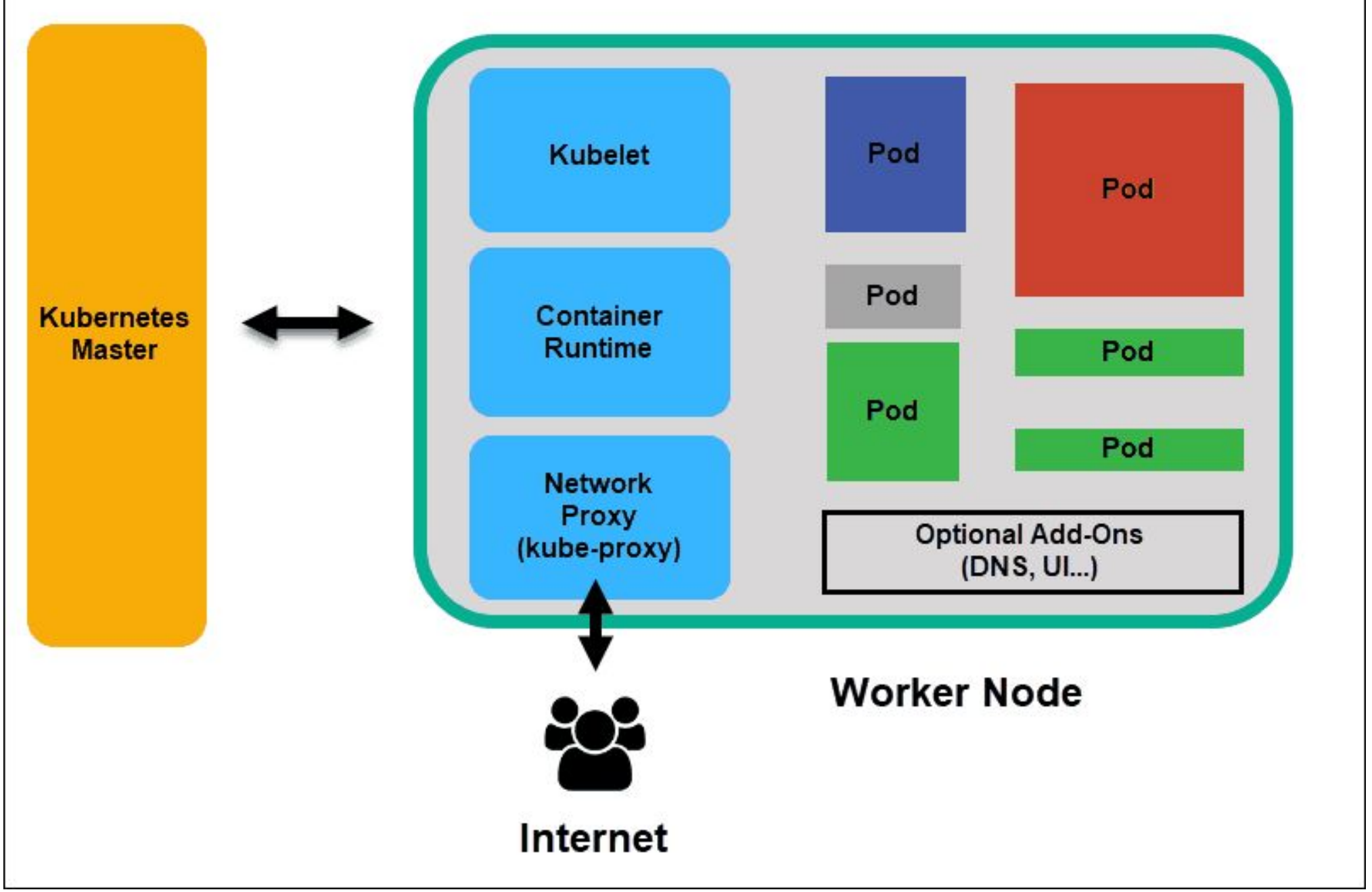
An open source orchestration system for managing containerized workloads across a cluster of nodes.

History of Kubernetes

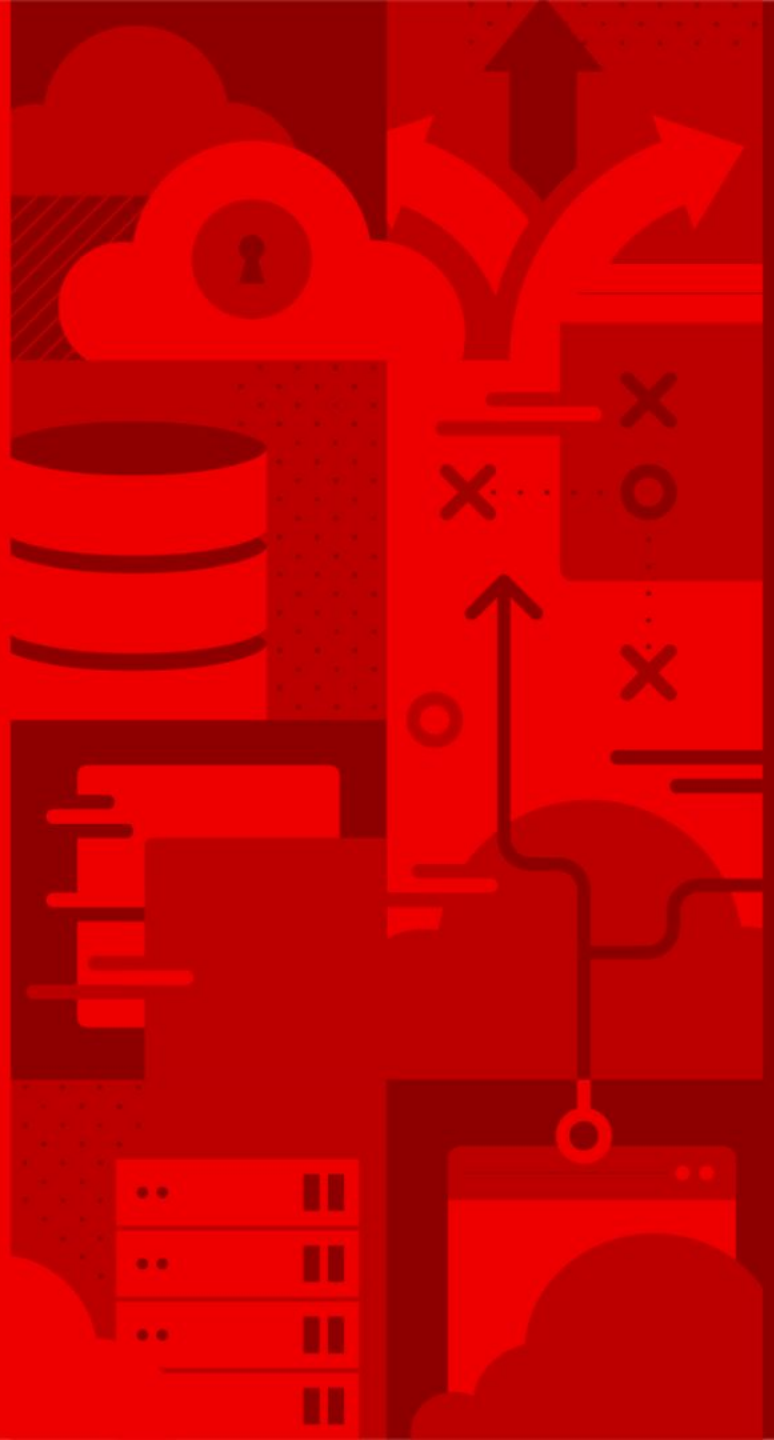
- Came from Google (announced in 2014)
- Influenced by Google's own cluster manager (Borg/Omega)
- Greek word for Helmsman or Governor
  - The captain of the container ship
- Written in GOLANG
- Google and the Linux foundation formed the CNCF (Cloud Native Computing Foundation) and donated the first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


Kubernetes Control Plane
kube-controller manager
cloud-controller manager
Cloud
kube-api-server
kubelet
kube-proxy
kubelet
kube-proxy
kubelet
kube-proxy
etcd
kube-scheduler
Kubernetes Nodes

What is



https://phoenixnap.com/kb/wp-content/uploads/2019/10/components-worker-node-kubernetes.png



Kubernetes basic building blocks

Red Hat

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

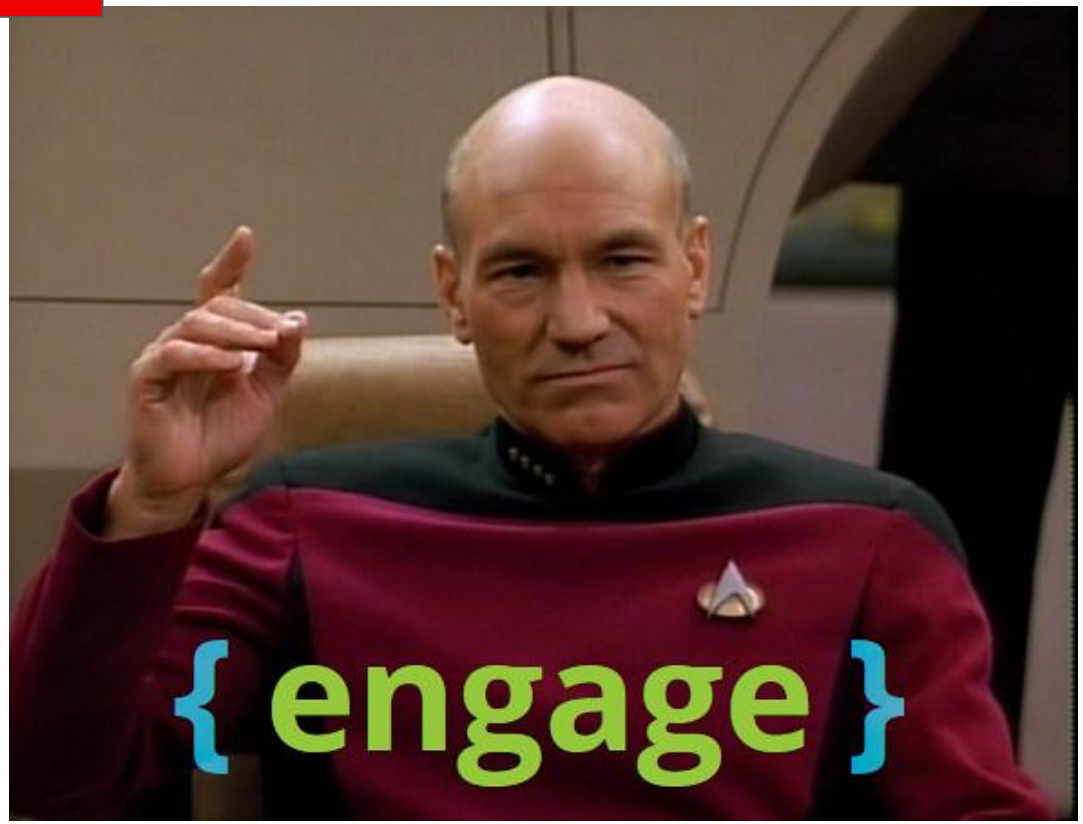
object1.yaml

```
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!

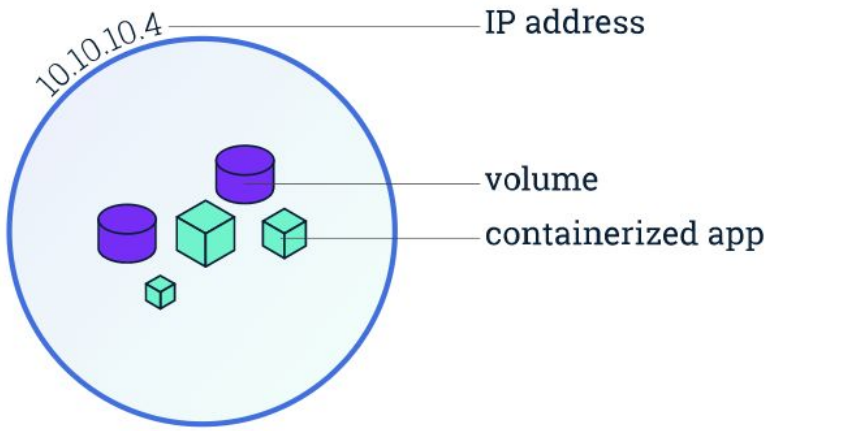


Pod

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.jso
```

```
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
```

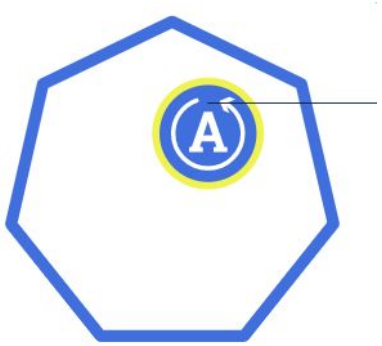
```
curl hello-k8s.<userX>:8080
```

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

DIY Kubernetes does not include all components that production needs

Do It Yourself Kubernetes - some assembly required

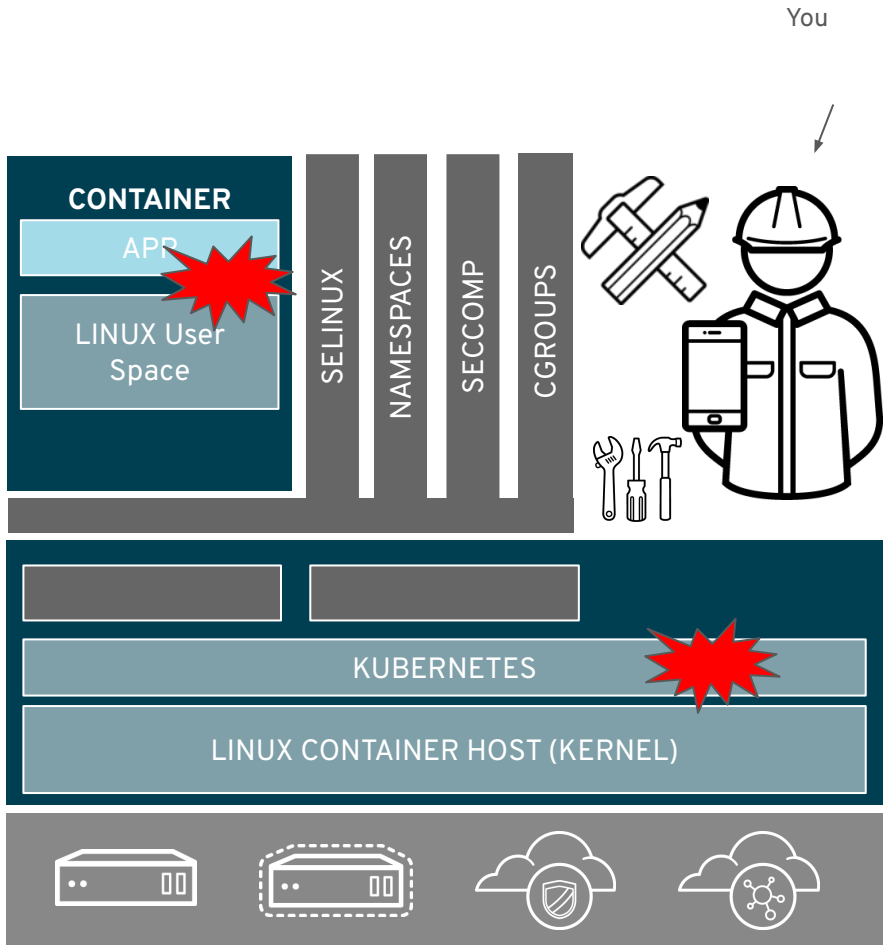
https://twitter.com/datamattsson/status/1275814200498151425

DIY: WHAT YOU NOW OWN

If you want to run DIY Kubernetes…

You will now OWN:

- Building your full DIY Kubernetes Stack
- Testing your DIY Kubernetes Stack
- Keeping up with Kubernetes (every 90 days)
- Updating add-on projects
- Patching OS Security CVEs (1 per week)
- Finding full-stack problems
- Fixing full-stack problems



All without any enterprise Technical Support

OCP vs DIY Kubernetes

	Red Hat OpenShift 4	kubernetes
Service mesh	✓ Istio, Jaeger, Kiali, Prometheus&Grafana	✗
Logging	✓ EFK	✗
Metrics	✓ Prometheus/Grafana	✗
Storage	✓ OpenShift Container Storage	✗
Network	✓ OVN	✗
Ingress	✓ Kubernetes Ingress/Routes	✓ Kubernetes Ingress
Ingress controller	✓ HA Proxy	✗
Egress	✓ Egress Router	✗
Authentication	✓ Kubernetes Auth/RH-IdM	✓ Kubernetes Auth
App isolation	✓ Kubernetes scheduler	✓ Kubernetes Scheduler
Infrastructure	✓ Bare metal, vSphere, KVM, OpenStack, AWS, GCP, Azure	✓ BYO Linux
Infra automation	✓ Ansible/Operators	✗
Infra management	✓ Admin Console	✗
Operating system	✓ RHEL or RHEL CoreOS	✗

OCP vs DIY Kubernetes

	Red Hat OpenShift 4	kubernetes
Container orchestration	✓ Kubernetes	✓ Kubernetes
Container image	✓ OCI-compliant	✓ BYO OCI-compliant
Container runtime	✓ CRIO	✓ BYO OCI-compliant engine
Container build	✓ RHCC/S2I/containerfile	✗
Container registry	✓ Quay/OSS docker registry	✗
Container scanner	✓ Clair	✗
CI/CD automation	✓ OCP Pipelines/Tekton	✗
IDE	✓ Che/Code Ready workspaces	✗
Web UX	✓ Web console (admin and developer perspectives)	✓ Web console
CLI UX	✓ oc/odo/kubectl	✓ kubectl
Service catalog	✓ Operators	✗
Secrets management	✓ Kubernetes Secrets	✓ Kubernetes Secrets

Red Hat

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

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat