Introduction to Kubernetes

Marky Jackson, Sr. Software Engineer AT&T ~ CCIE 4736

About Marky

- Stanford
- Yahoo
- Symantec
- AT&T
- Kubernetes Developer Advocate
- Jenkins Community Community Contributor
- Lover of the San Francisco Giants

Introduction

- Kubernetes is open source cluster manager for container orchestration, management and resilience

- Also known as K8s ~ k followed by 8 chars and s

- K8s automates the orchestration of docker, rtk and etc

- Greek word for Pilot, or Helmsman

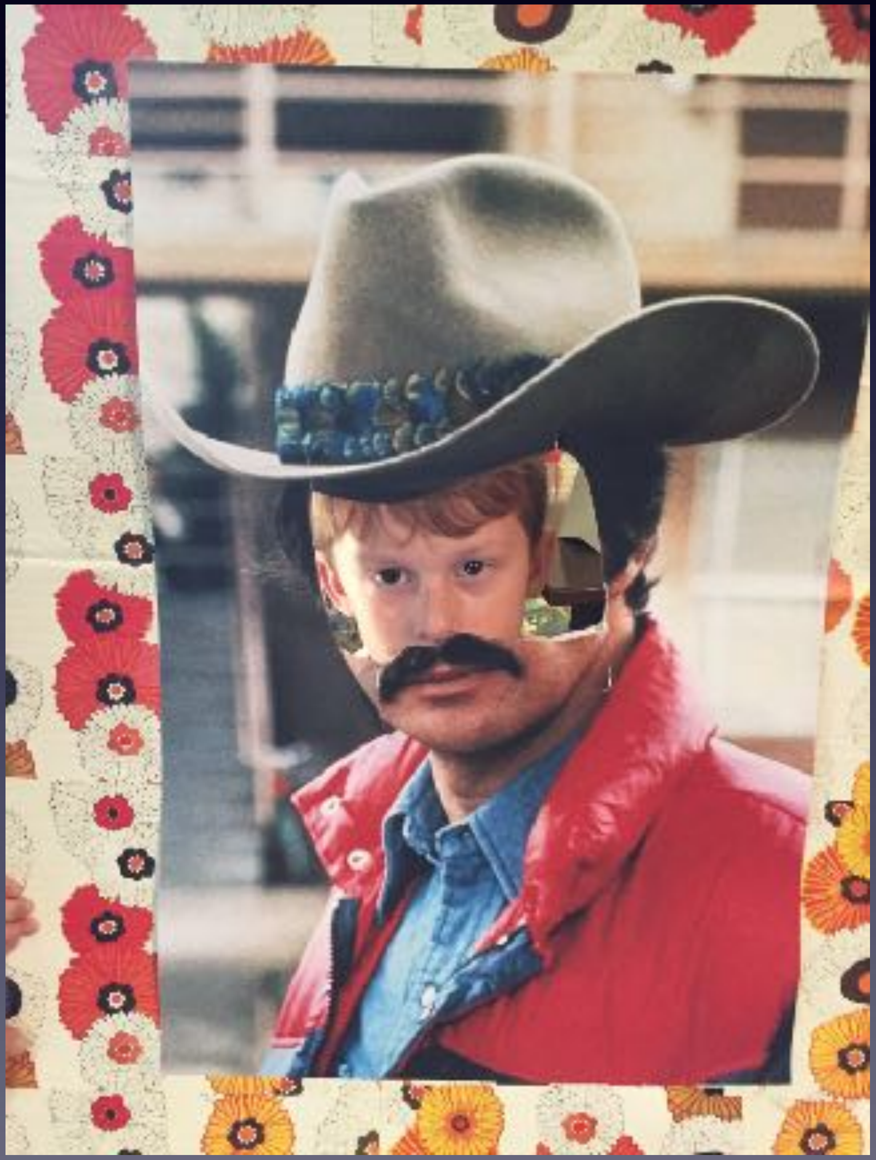
Birth

- K8s was built from Google Borg which manages Google datacenter

- Open sourced for Cloud native Foundation (CNCF)

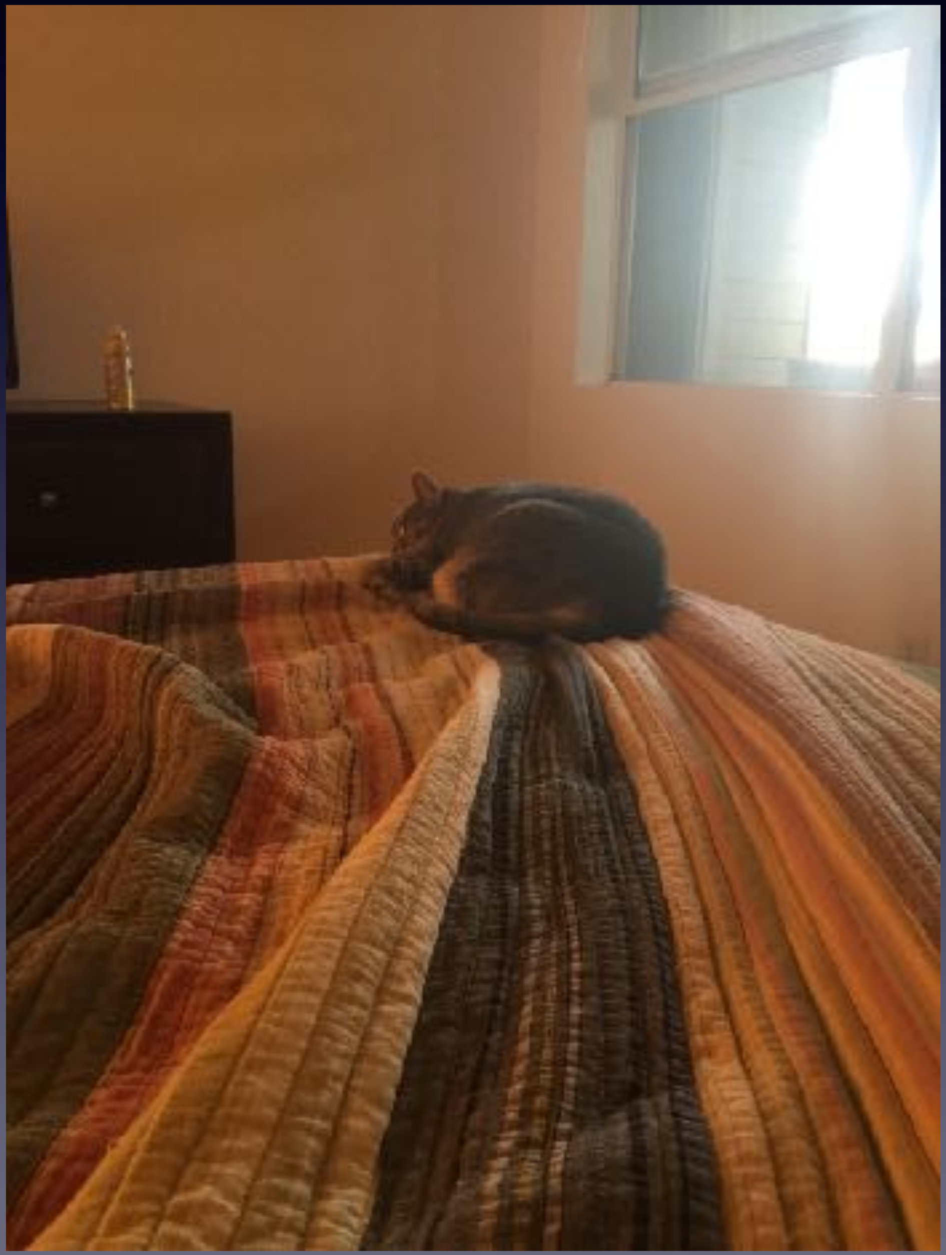
Features

- Automates the life of a container
- Service discovery and load balancing
- Self healing capability
- Works with Docker and Coreos Rkt
- Works in private/public/hybrid clouds
- Storage orchestration

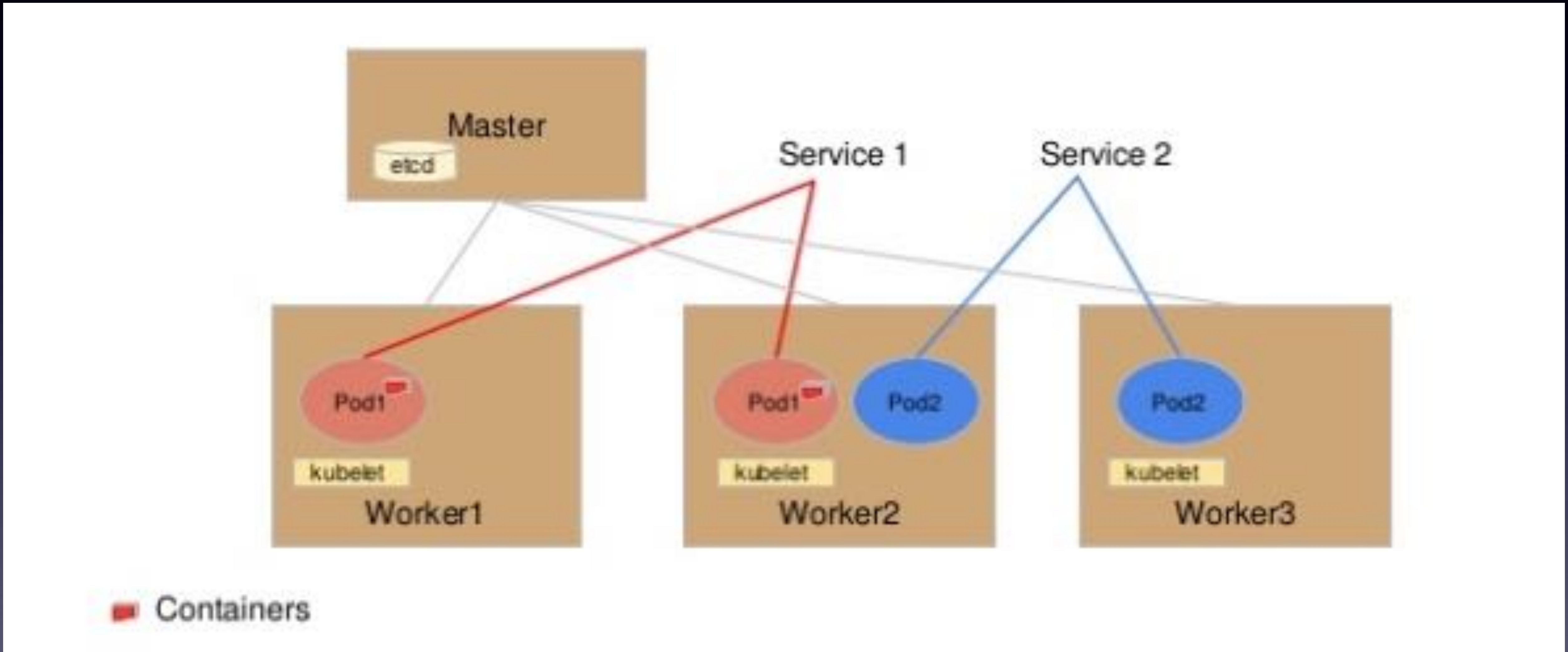


Main Concept

- Cluster
- Pods
- Replication
- Services



High Level Architecture



Master & Workers

- Master maintains the state of the cluster on etc backend
- There is a single point of entry for al clients and API calls to manage the cluster and other components like pods/services
- Parts of the master
    - API Server
    - Registry (node, pods, services)
    - Etcd storage objects

Cluster

- Logical group of compute, storage and networking

- Network is super important in supporting east-west traffic patterns

Pods

- Tiny deployable part of computer - It views the logical view of the application
- Group of containers that are co-located and scheduled to run on a node
- Pods are ephemeral but volumes are persistent
- Pods can be labeled which will allow tagging and grouping to enable replication
- Containers inside the pods share namespace and are accessible using the same IP
- Pods are defined by a YAML base manifest file

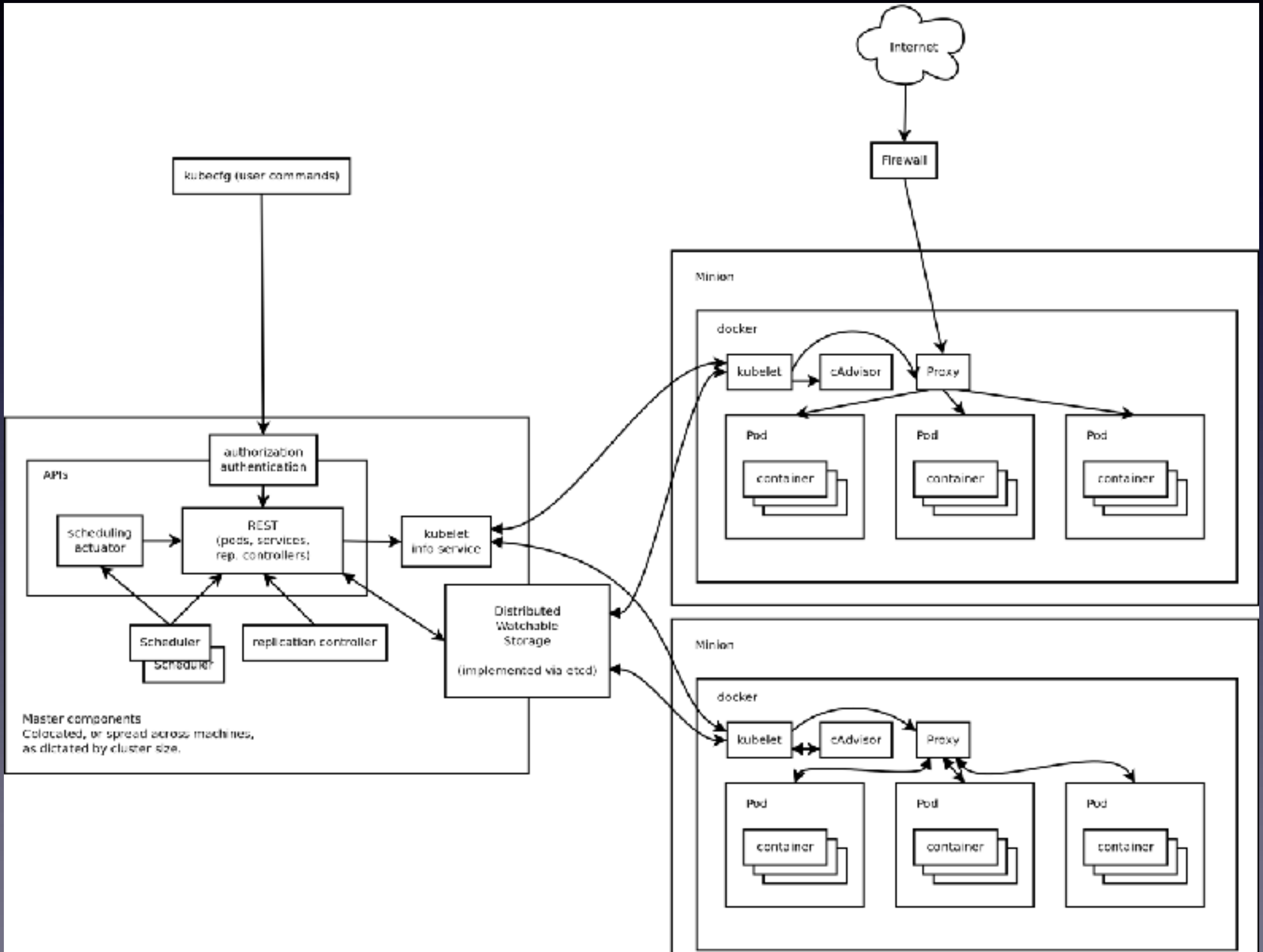
Replication Controller

- Replicates the pods across a node
- Elasticity - You can scale up and down
- Assist in rolling updates/ upgrades

Services

- Offers persistent endpoints of pods
- Abstraction that associates policies between pods
- Similar to the virtual IP concept of load balancing
- Great way to achieve micro services
- Helps with services discovery

Kubernetes Architecture



Scheduler

- Locates a node to place the given pod (based on PodSpec)
- Policy Driven
    - Fit Predicate (filtering rules)
    - PriorityFunction
- Extensible scheduling policies
- Kube-scheduler runs on the master node
- Once a pod is bound to a node, kubelet handles it

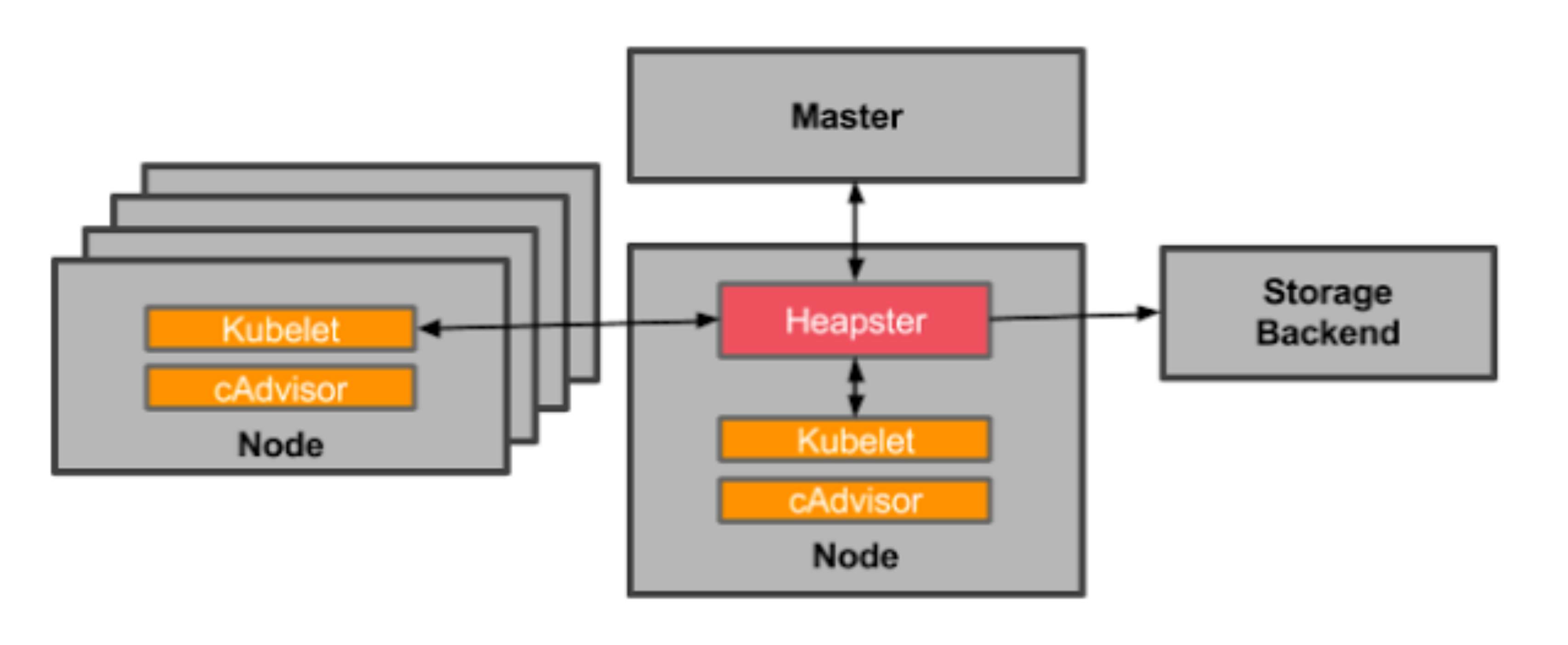
Kubelet

- This is an agent running on each worker node
- Manages the lifecycle running on each pod
- Registers with the master
- Health updates of the node to the master
- Runs etcd client

API Server

- Service REST Operation
- Interacts with etcd to frontend the cluster state
- Every component makes use of the api server to get cluster state
- Kubectl provides cli functionality

Resource Monitoring



cAdvisor

- https://github.com/google/cadvisor
- Resource usage and performance analysis agent
- Integrated into kubelet agent
- Native support for docker containers
- Auto-discovery of containers in a node
- Collection of metrics - cpu, mem, network etc.
- Overall machine usage for node (workers)

Heapster

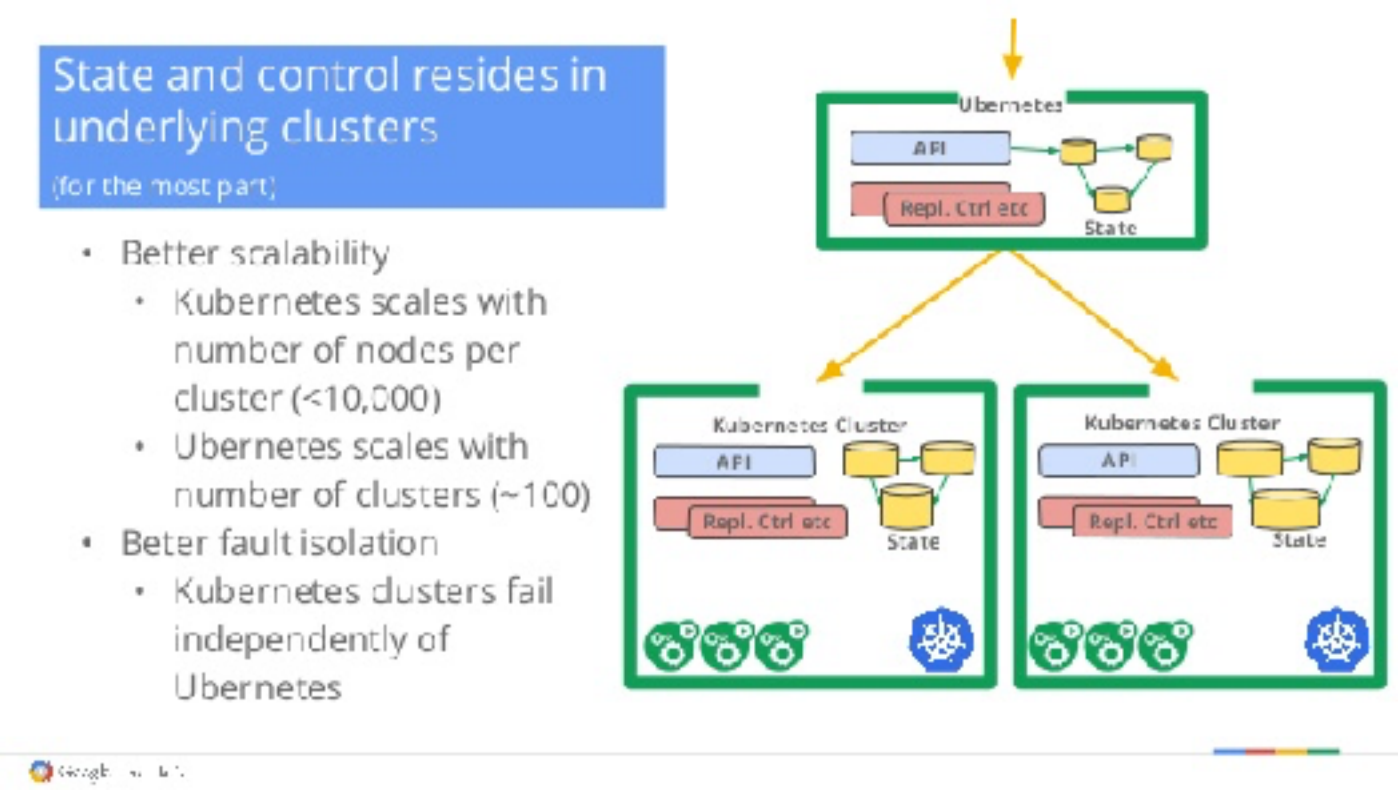
- Cluster-wide aggregate of monitoring and data events
- Runs as a separate pod
- The heapster pod discovers all nodes in the cluster
- Queries usage info from nodes kubelet (cAdvisor)
- Aggregation - pod, label level
- Ability to store different sinks - logs, influxdb, kafka, elastic search etc.

Bells and Whistles

- Minikube - Single node cluster that run on a VM on your laptop
- Helm - A Package manager, collection of ‘charts’

Ubernetes

- Federation of Kubernetes cluster
- Allows switching workloads between clusters
- Avoids vendor lock-in



State and control resides in underlying clusters

(for the most part)

- Better scalability
  - Kubernetes scales with number of nodes per cluster (<10,000)
  - Ubernetes scales with number of clusters (~100)
- Beter fault isolation
  - Kubernetes clusters fail independently of Ubernetes

Questions

- Ping me at marky.r.jackson@gmail.com

Thank You