Kubernetes Architecture

Janakiram MSV
Analyst | Advisor | Architect
Janakiram & Associates
Google Developer Expert

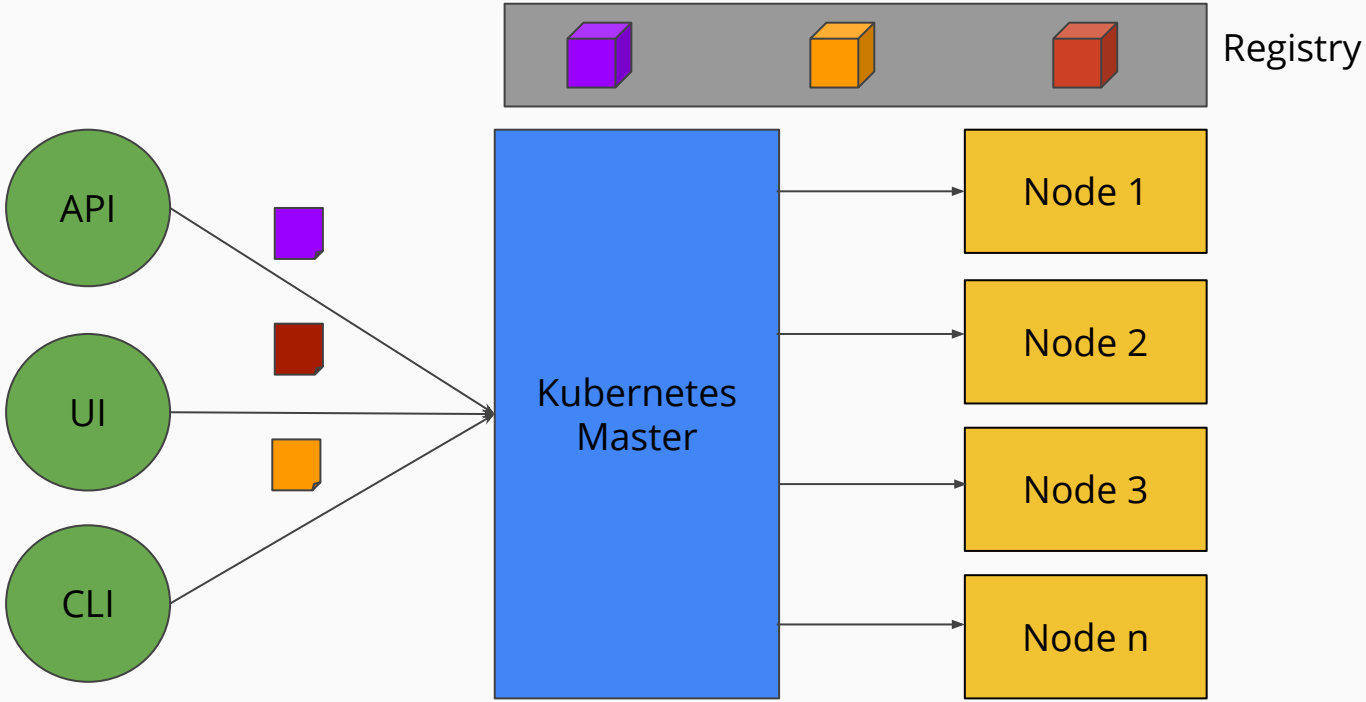
Announcing
The Kubernetes Webinar Series

- Monthly webinar focused on helping you get started with Kubernetes
- Delivered by the experts from the container ecosystem
- Each webinar introduces a concept to accelerate your learning
- No specific prerequisites
  - Basic understanding of Docker will help
- All attendees will get $100 Google Cloud Platform Credits
- Win free annual and monthly subscriptions of MAPT eBook library
- The recording of this webinar will be available at Kubernetes.live

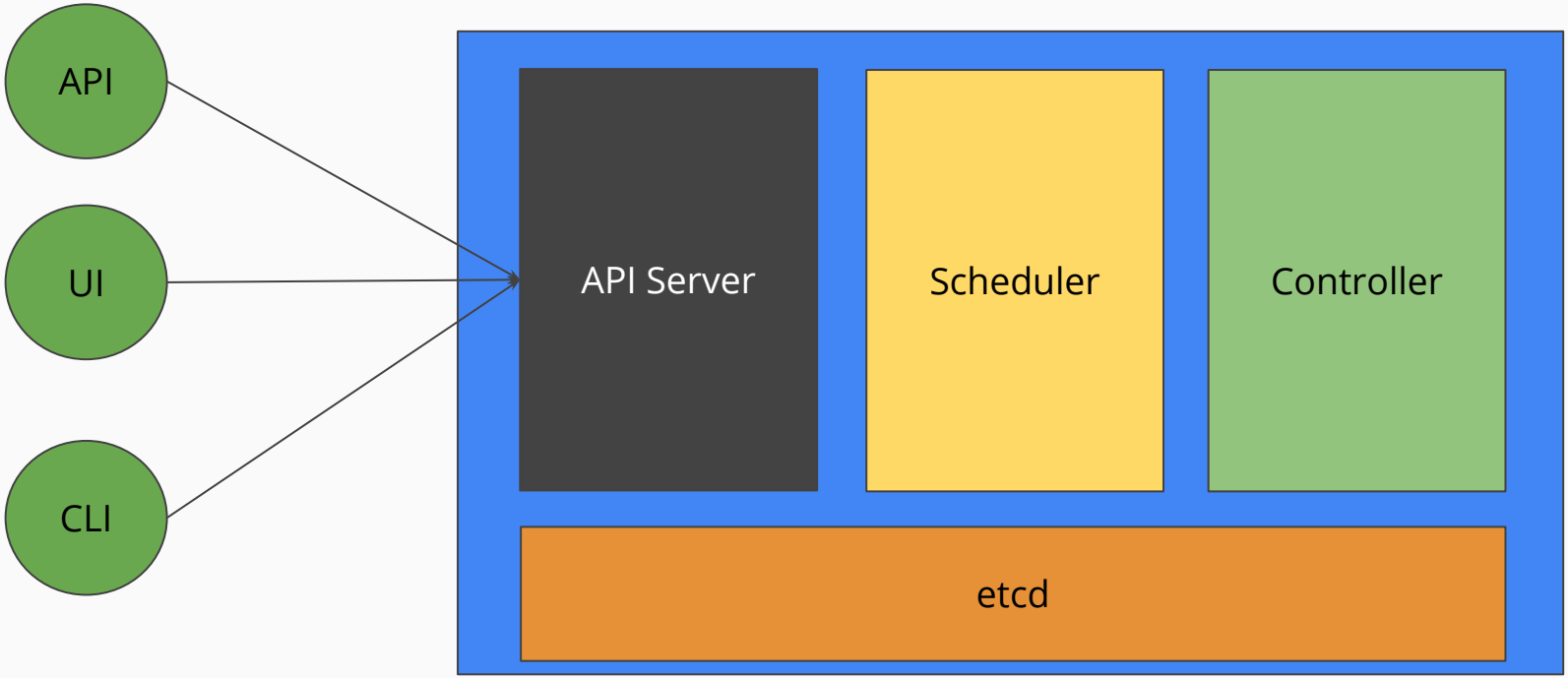
Objectives

- A closer look at Kubernetes cluster
- Master components
- Node components
- Pods
- Labels & Selectors
- Replication Controllers
- Services

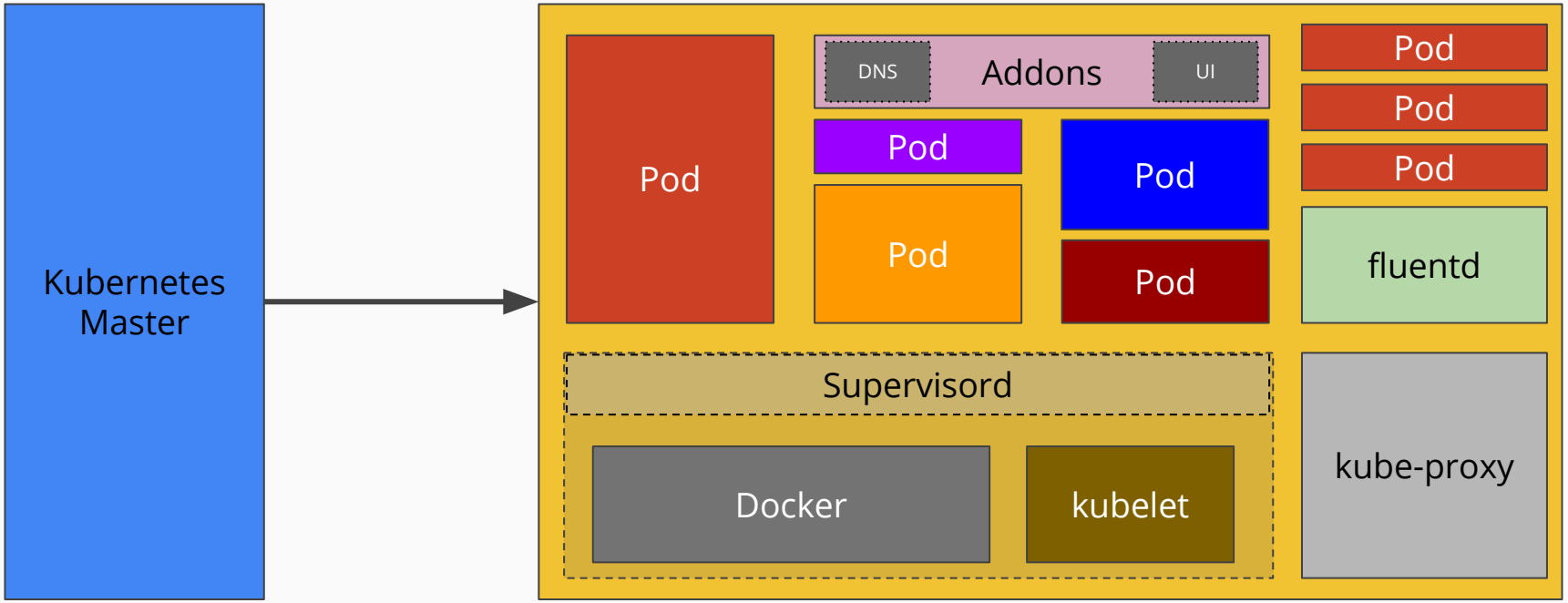
Kubernetes Architecture



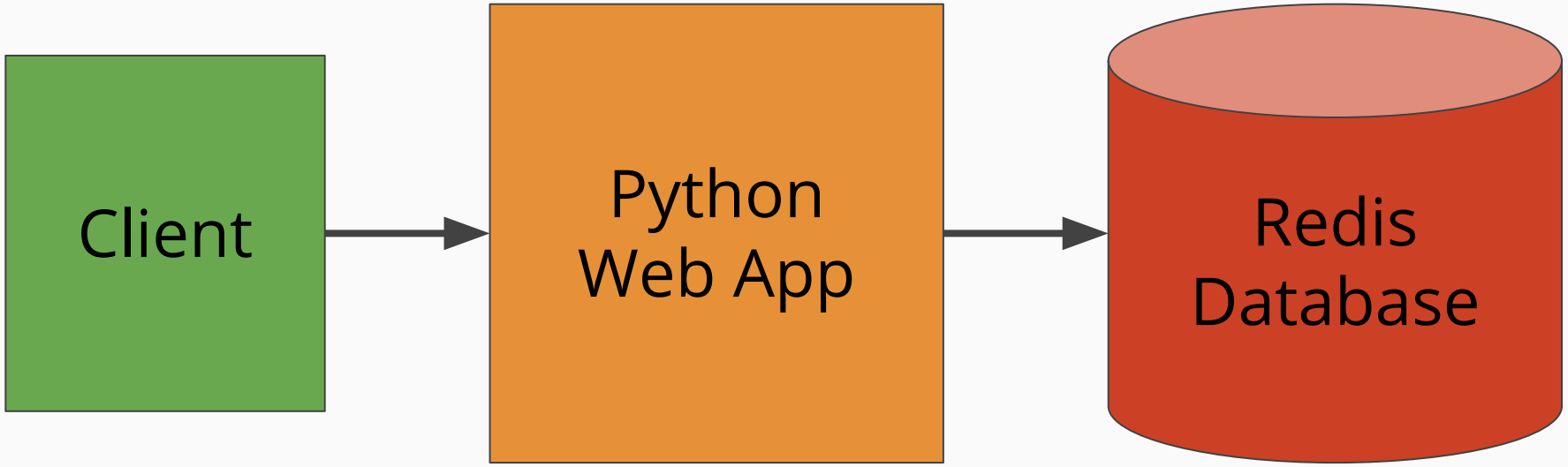
Kubernetes Master



Kubernetes Node



A Simple Containerized Application



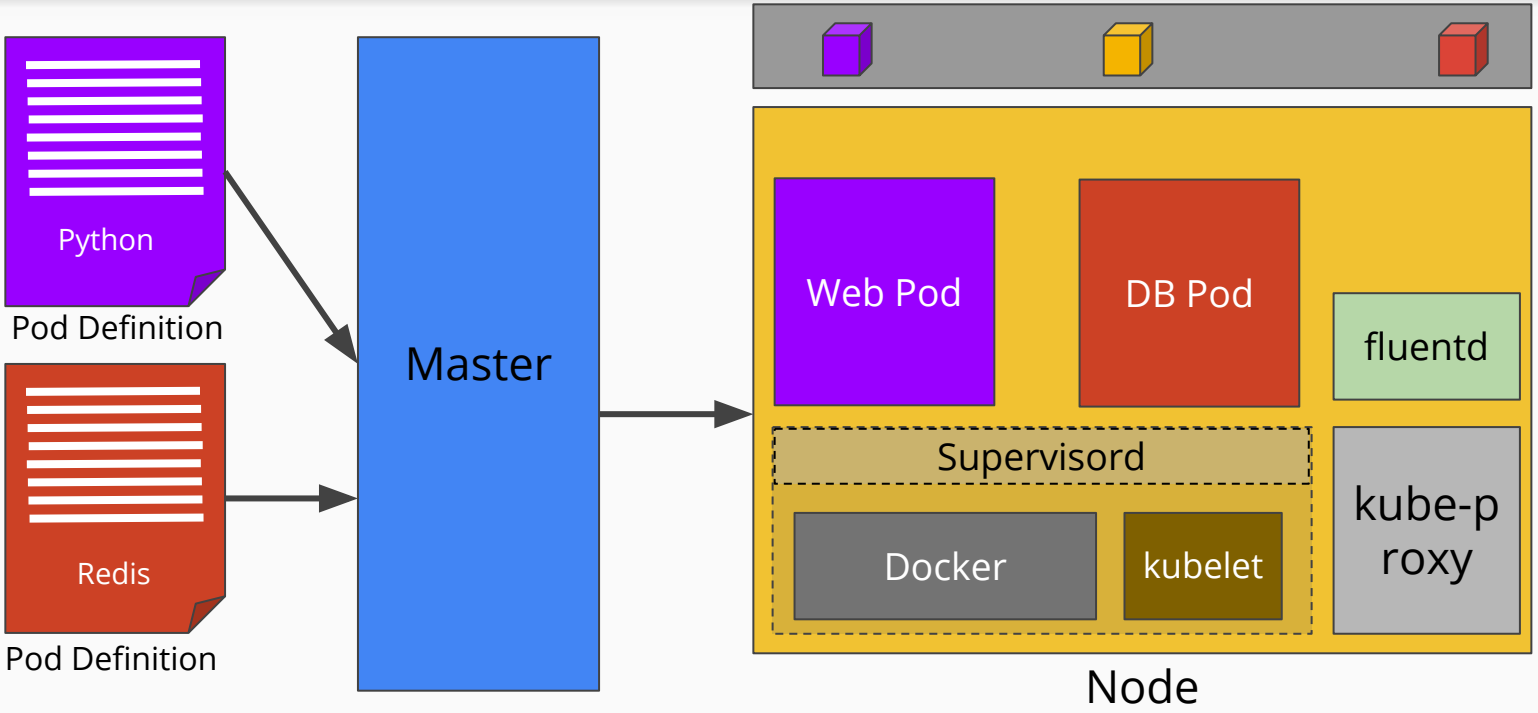
Kubernetes Pod

- Group of one or more containers that are always co-located, co-scheduled, and run in a shared context
- Containers in the same pod have the same hostname
- Each pod is isolated by
  - Process ID (PID) namespace
  - Network namespace
  - Interprocess Communication (IPC) namespace
  - Unix Time Sharing (UTS) namespace
- Alternative to a VM with multiple processes

Labels & Selectors

- Key/value pairs associated with Kubernetes objects
- Used to organize and select subsets of objects
- Attached to objects at creation time but modified at any time.
- Labels are the essential glue to associate one API object with other
  - Replication Controller -> Pods
  - Service -> Pods
  - Pods -> Nodes

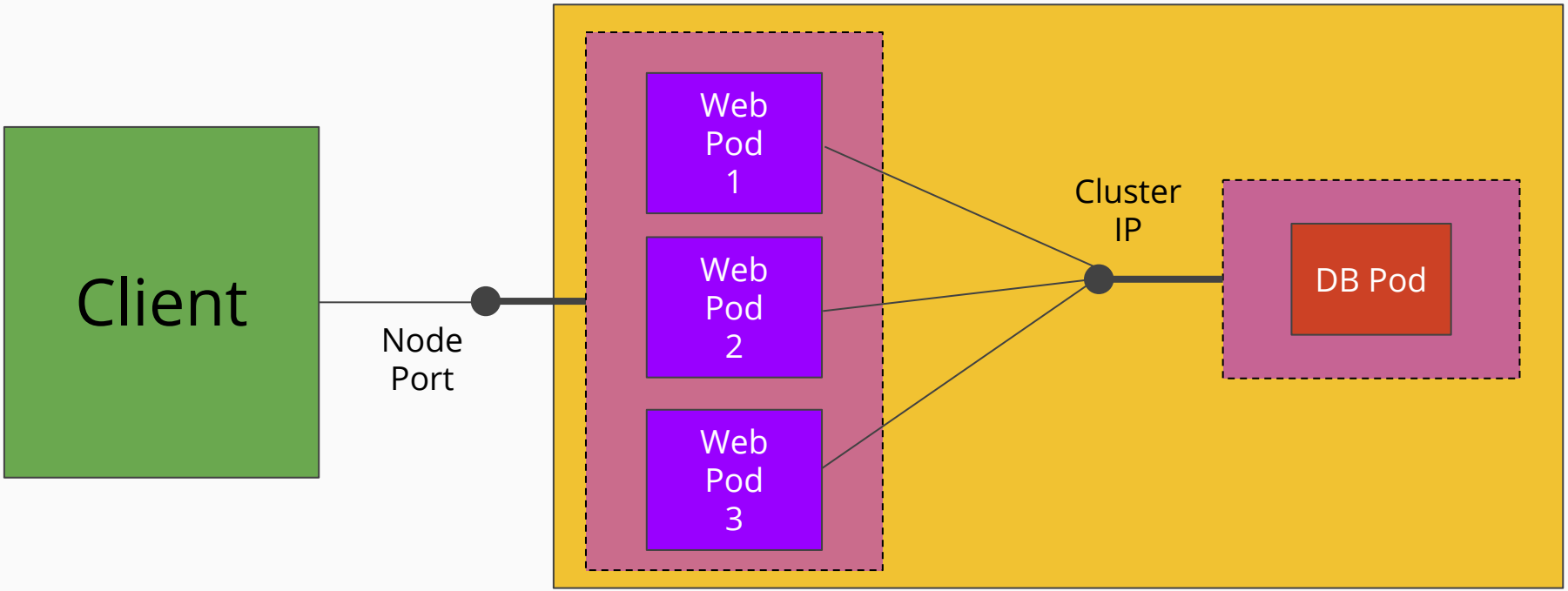
Deploying a Pod



Services

- An abstraction to define a logical set of Pods bound by a policy by to access them
- Services are exposed through internal and external endpoints
- Services can also point to non-Kubernetes endpoints through a Virtual-IP-Bridge
- Supports TCP and UDP
- Interfaces with kube-proxy to manipulate iptables
- Service can be exposed internal or external to the cluster

Exposing Services



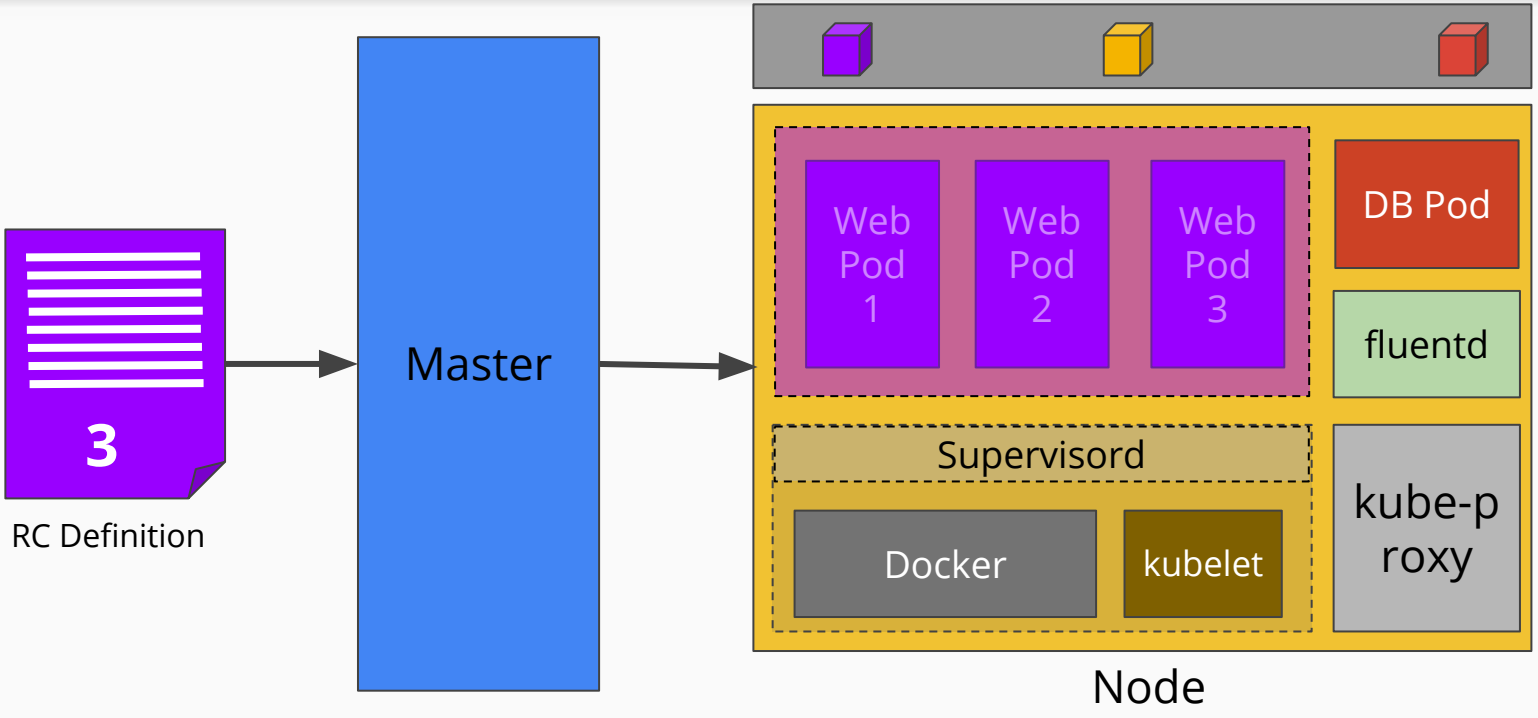
Demo

Creating Pods & Services
Python Web App & Redis DB

Replication Controller

- Ensures that a Pod or homogeneous set of Pods are always up and available
- Always maintains desired number of Pods
  - If there are excess Pods, they get killed
  - New pods are launched when they fail, get deleted, or terminated
- Creating a replication controller with a count of 1 ensures that a Pod is always available
- Replication Controller and Pods are associated through Labels

Scaling Pods with Replication Controller



Demo

Scaling Pods with Replication Controller

Summary

- Kubernetes Master runs the API, Scheduler and Controller services
- Each Node is responsible for running one or more Pods
- Pods are the unit of deployment in Kubernetes
- Labels associate one Kubernetes object with the other
- Replication Controller ensures high availability of Pods
- Services expose Pods to internal and external consumers

Upcoming Webinar

A Closer Look at Kubernetes Pods

9 AM PST, 26th October

This session will discuss the best practices of creating, deploying, and managing Pods. We will take a closer look at the storage and networking architecture of containers running in the same Pod.

Thank You!

Send your Feedback / Questions / Comments
info@kubernetes.live