Persistent

Introduction to Kubernetes

What Docker Is... and What it Isn't

- Docker and its ecosystem are great for managing images and running containers in a specific host.
- Docker alone can't manage your containers across multiple nodes,
- Docker can’t alone schedule and manage tasks to be completed across your cluster.
- In order to manage a Docker workload on a distributed cluster, you need to bring some other players into the mix.
- If you want Docker containers managed across vast fleets of servers and infrastructure, then you use Docker orchestration tools like Kubernetes.

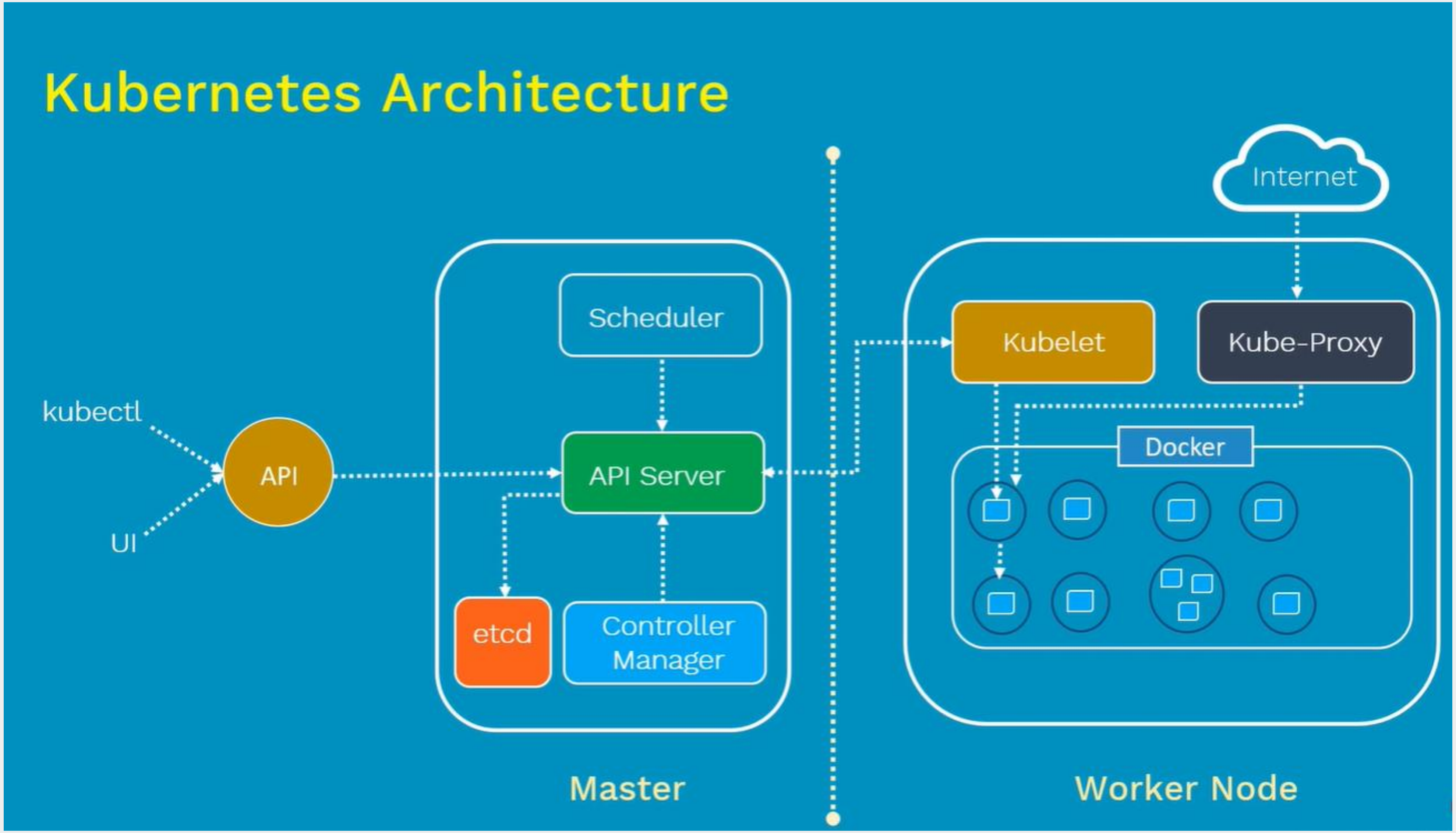
Kubernetes

- Developed by Google, Kubernetes is essentially a cluster manager for Docker.
- Using Kubernetes, you can schedule and deploy any number of container replicas onto a node cluster.
- Kubernetes will take care of making decisions like which containers go on which servers for you.
- Kubernetes is a solution for overseeing and managing multiple containers at scale, rather than just working with Docker on a manually-configured host.

Why Kubernetes?

- Kubernetes was designed to make working with containers on Google Compute Engine easier.
- But you need not be running GCE.
- Kubernetes packages all necessary tools -- orchestration, service discovery, load balancing -- together in one nice package.
- Kubernetes is a solution for overseeing and managing multiple containers at scale, rather than just working with Docker on a manually-configured host.

Key Components of Kubernetes



Kubernetes key components…

- Master: the managing machine, which oversees one or more minions.
- Minion: a slave that runs tasks as delegated by the user and Kubernetes master.
- Pod: an application (or part of an application) that runs on a minion. This is the basic unit of manipulation in Kubernetes.
- Replication Controller: ensures that the requested number of pods are running on minions at all times.
- Label: an arbitrary key/value pair that the Replication Controller uses for service discovery
- kubecfg: the command line config tool
- Service: an endpoint that provides load balancing across a replicated group of pods

How to Use Kubernetes

- Pull down and start the Kubernetes services, and you're ready to build your first pod.
- Interacting with Kubernetes is quite simple, even outside of the context of Google Compute Engine.
- Kubernetes provides a RESTful API in order to manipulate the three main resources: pods, services, and replication Controllers.
- The basic steps for running Kubernetes are provided in the below links.
  - https://www.ctl.io/developers/blog/post/what-is-kubernetes-and-how-to-use-it/
  - http://kubernetes.io/docs/hellonode/

Reference Material : Websites & Blogs

- https://www.ctl.io/developers/blog/post/what-is-kubernetes-and-how-to-use-it/
- http://kubernetes.io/docs/hellonode/

Key Contacts

Docker Interactive

Dattatray Kulkarni

dattatray_kulkarni@persistent.co.in

Asif Immanad

asif_immanad@persistent.co.in

Vice President

Shubhangi Kelkar

shubhangi_kelkar@persistent.co.in

Thank you!

Persistent University