Kubernetes (K8s)

(Container Orchestration System)

BUILDING EXCELLENCE

Contents

What is Container?

Docker Overview

Kubernetes Overview

Quick Docker Demo

Kubernetes Demo

SYNTEL

Everyone Loves Containers

What is Container?

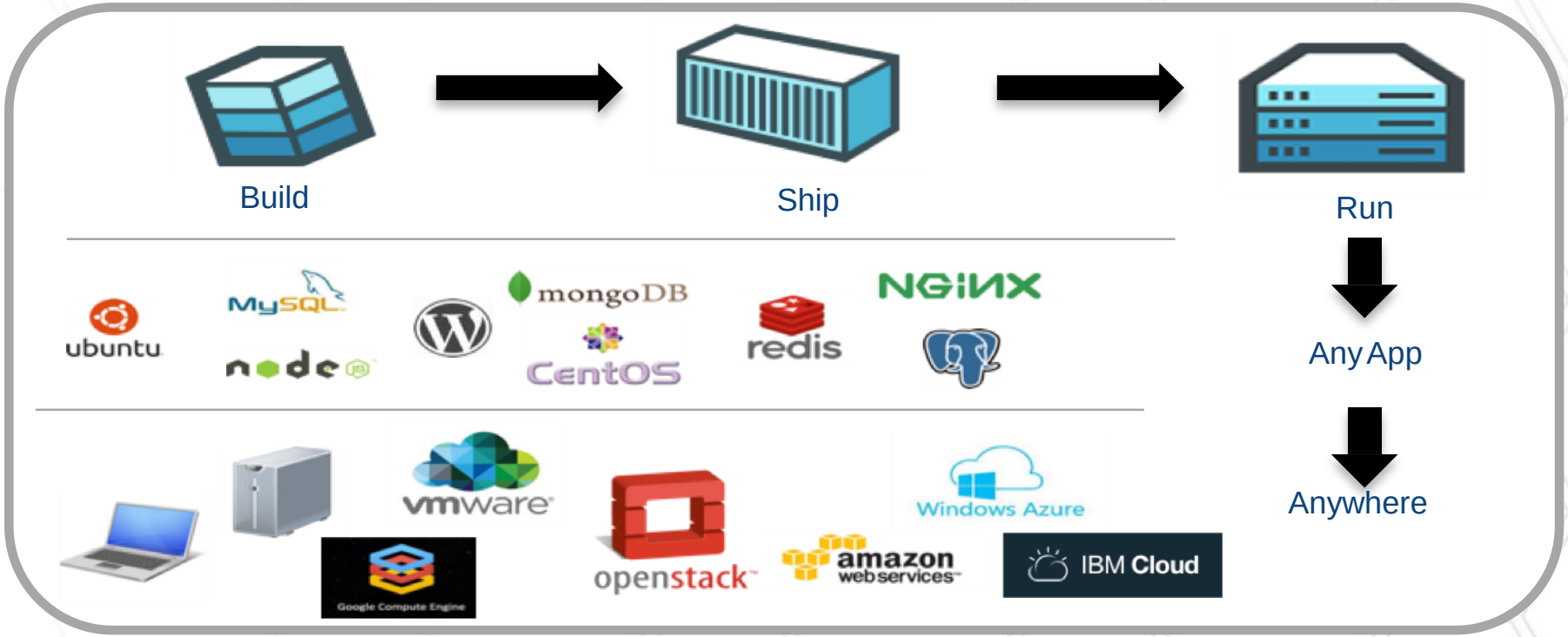
- A standard way to package an application and all its dependencies so that it can be moved between environments and run without changes
- Containers work by isolating the differences between applications inside the container so that everything outside the container can be standardized

Container Advantages

- Containers are portable
- Containers are easy to manage
- Containers provide “just enough” isolation
- Containers use hardware more efficiently
- Containers are immutable

Docker Mission

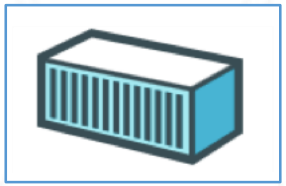
Docker is an **open platform** for building distributed applications for **developers** and **system administrators**



Docker Components

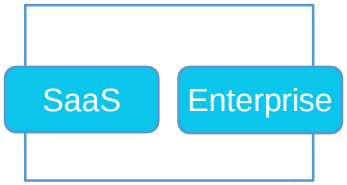
Image

- A read-only snapshot of a container stored in Docker Hub to be used as a template for building containers



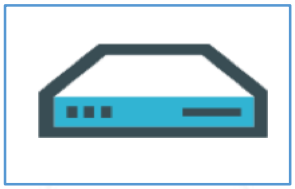
Container

- The standard unit in which the application service resides or transported



Docker Hub/Registry

- Available in SaaS or Enterprise to deploy anywhere you choose
- Stores, distributes, and shares container images



Docker Engine

- A program that creates, ships, and runs application containers
- Runs on any physical and virtual machine or server locally, in private or public cloud
- Client communicates with Engine to execute commands

Containers

Everyone’s container journey starts with one container….

Containers

At first the growth is easy to handle….

Containers

Container Needs

Health checks – up and running? How to restart?

Discovery – access containers

Communication – containers talk to each other

Security – sensitive data, authorization

Isolation – keep jobs separate

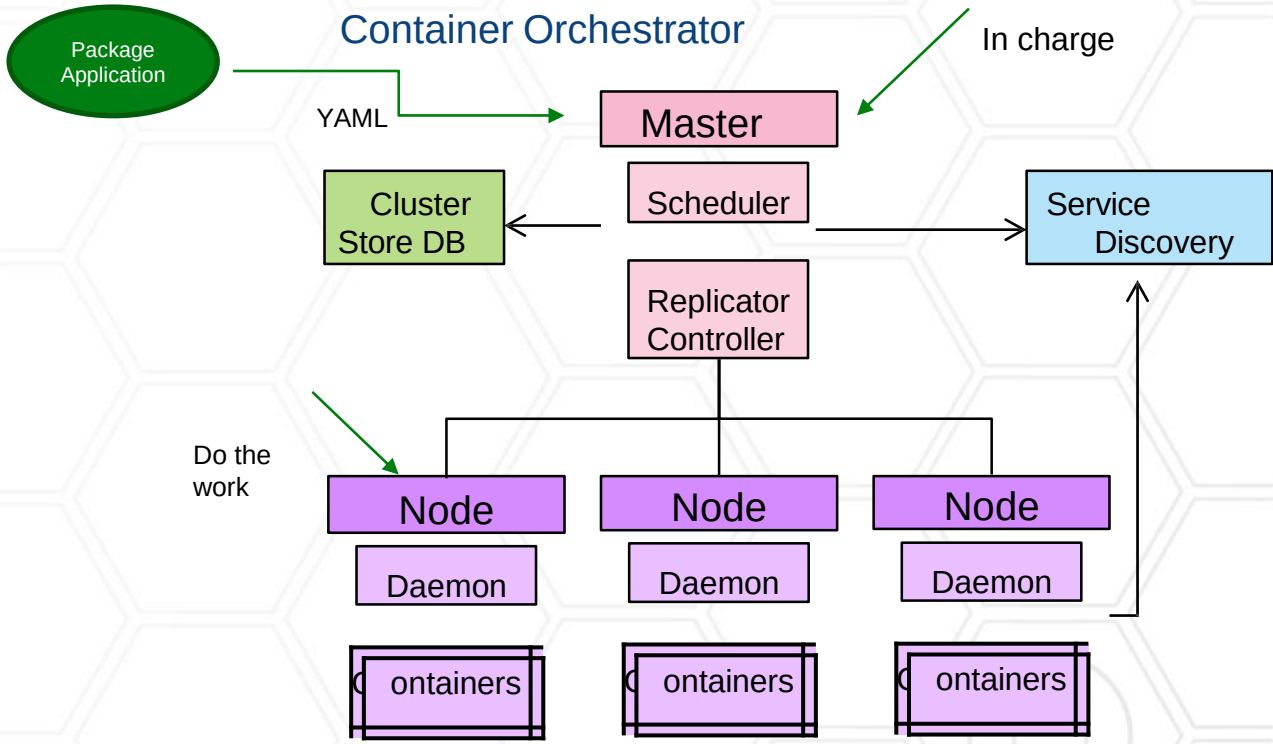
Scheduling – when should my jobs run? Lifecycle?

Scalability – make my jobs bigger/smaller

****Leads to great complexity**

What is Container Orchestration?

- Container orchestration
  - Cluster management
  - Scheduling
  - Service discovery
  - Replication
  - Health management



What is Kubernetes (K8s)?

Open-source automated deployment, scaling & management of containerized apps

- **Based on Google's infrastructure**
- **Problem:** How do I manage applications at scale?

– **The application:** How to build, package, distribute

– **The infrastructure:** How to make it scalable (efficiently)

– **The evolution:** How to handle your evolving code

- **Solution: Use Docker + Kubernetes**

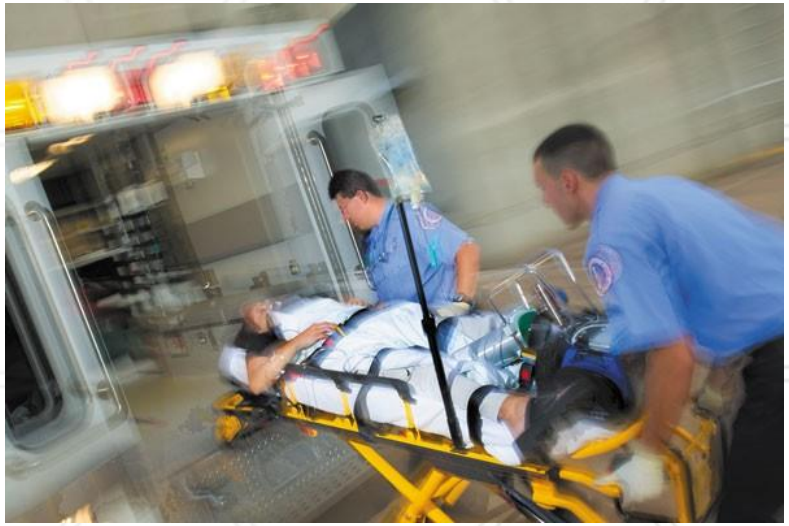
– **Docker:** Containers

– **Kubernetes:** Container management

- **Manage applications, not machines!**

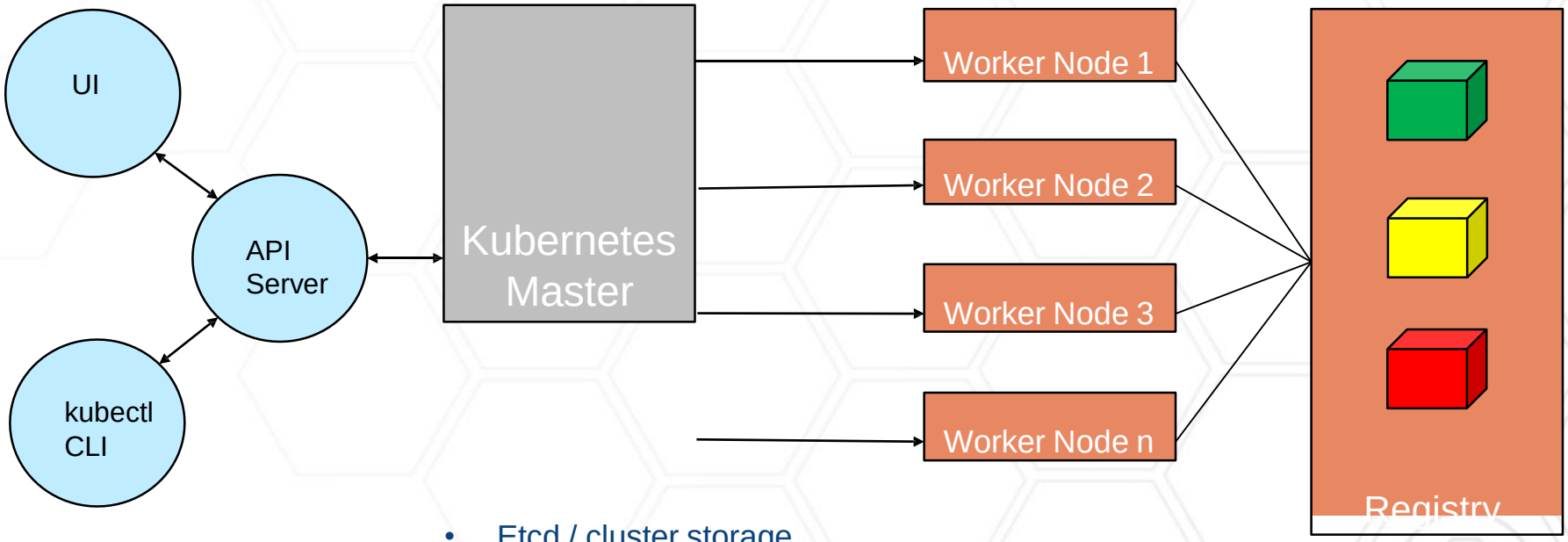
What is Kubernetes?

- Container orchestrator
- Manage applications, not machines
- Designed for extensibility
- Open source project managed by the Linux Foundation



Kubernetes Architecture

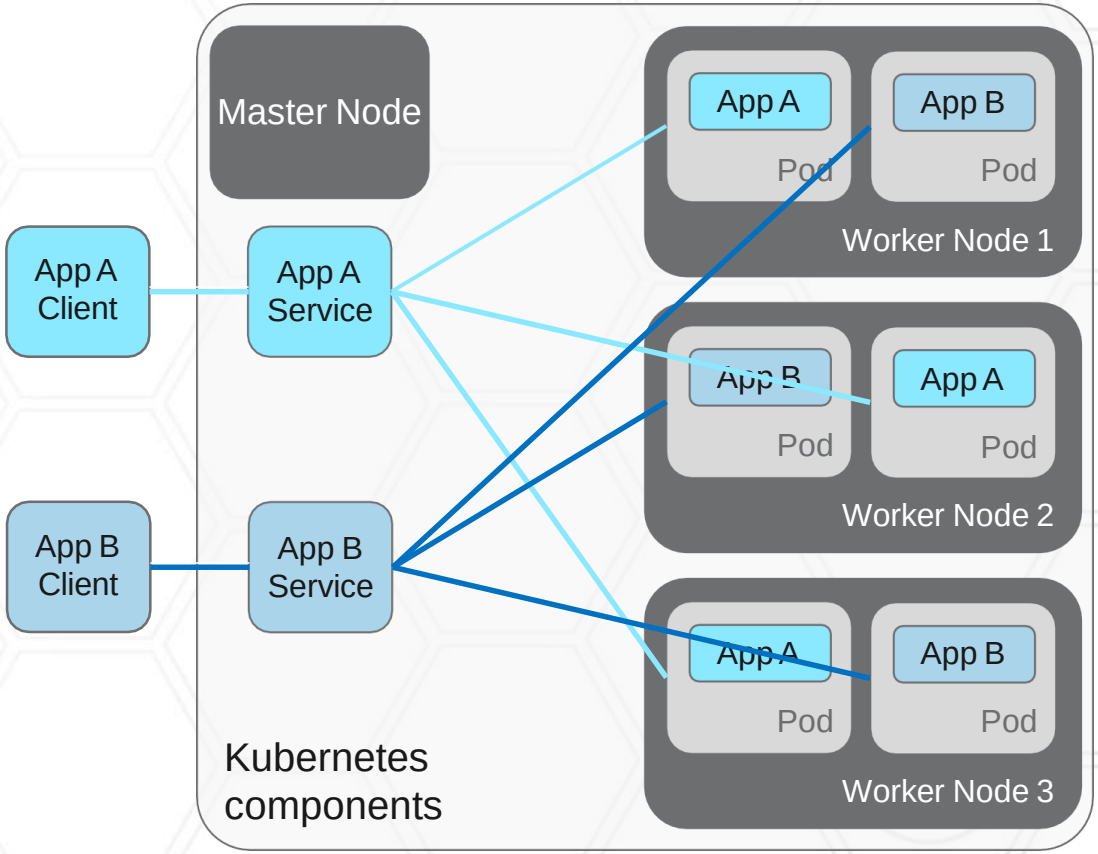
Desired state management



- Etcd / cluster storage
- API Server
- Replication Controller
- Scheduler
- Service Discovery

Kubernetes Architecture: Workloads

- Kubelet
  - Main Kubernetes node agent
  - Communicates with master
  - Instantiates pods
- **Kube Proxy (Assign IP address)**
- Container engine
  - Packaging of an app
- Pod
  - Unit of deployment
- Service
  - Fixed endpoint for 1+ pods



Common Docker Commands

Docker Commands to build and run images:

Build images for spring boot app using local Dockerfile:

mvn package docker:build

To list all docker local images:

docker images

docker container run —name containername -d -p 8080:8000 imagename

Now access the application using:
localhost:8080

Docker container stop container name:

docker container stop container name
docker rm container name
docker rmi image name

To open shell inside docker container:
docker exec -it container name sh

To persist data on host file system use docker volumes:

docker container run –d –-name container_name -p 8080:8000 –v myvol:/usr/share/text

docker container exec –it container_name sh
#cd /usr/share/text
#echo “This is my text file”
#exit … remove container file/dir still persist

Common Kubernetes Commands

Kubernetes Local development environment :

minikube start

kubectl config use-context minikube

Launch the dashboard:

minikube dashboard

Deploy a docker application to the kubernetes cluster through command line:

kubectl run myspringdemo --image=ganeshdockerjava/docker-spring-example --port=8000

- **Exposing the application through service using command line:**

  kubectl expose deployment myspringdemo --type=LoadBalancer --name myspringdemoservice

  To list the service:

  kubectl get service / svc

  To describe the service details:

  Kubectl describe services service-name

  Delete Deployment and Service:

  kubectl delete deployment deployment-name

  Kubectl delete service service-name