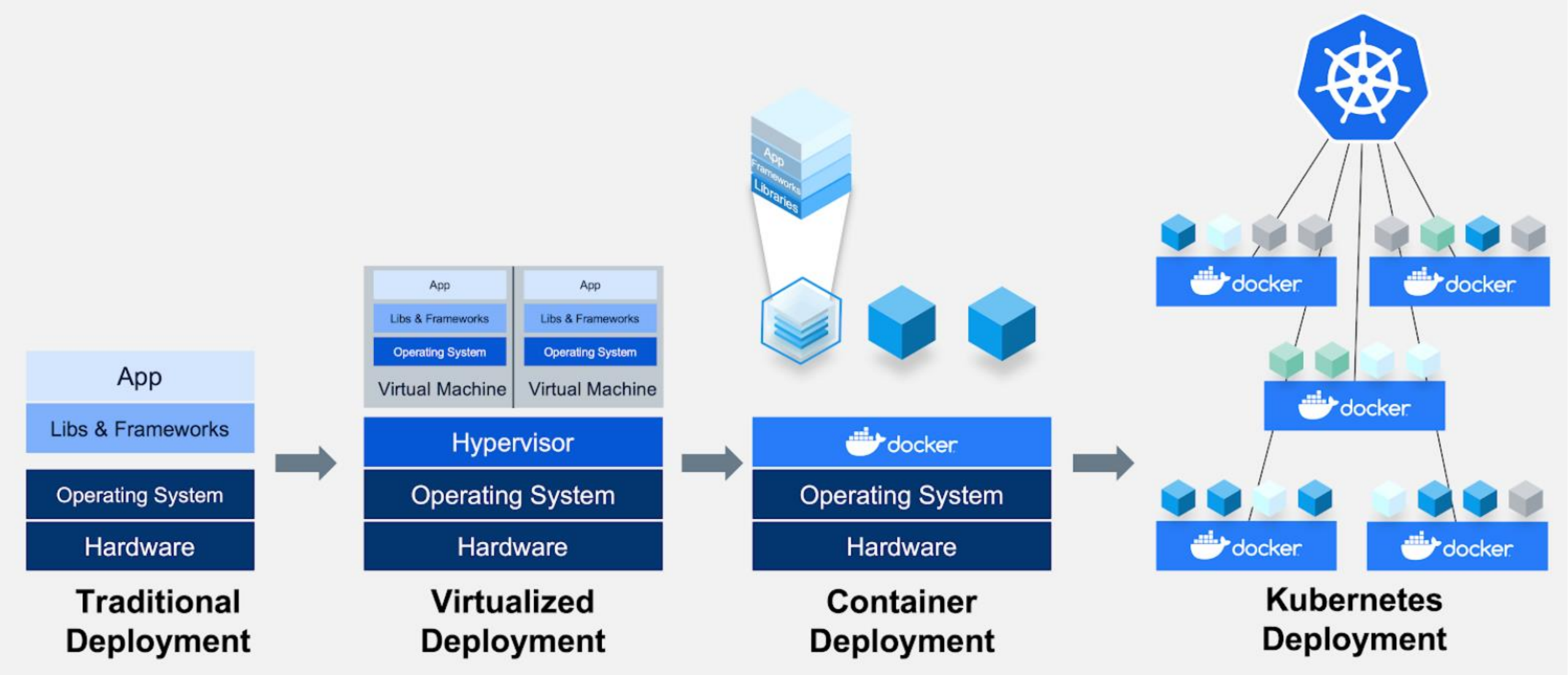
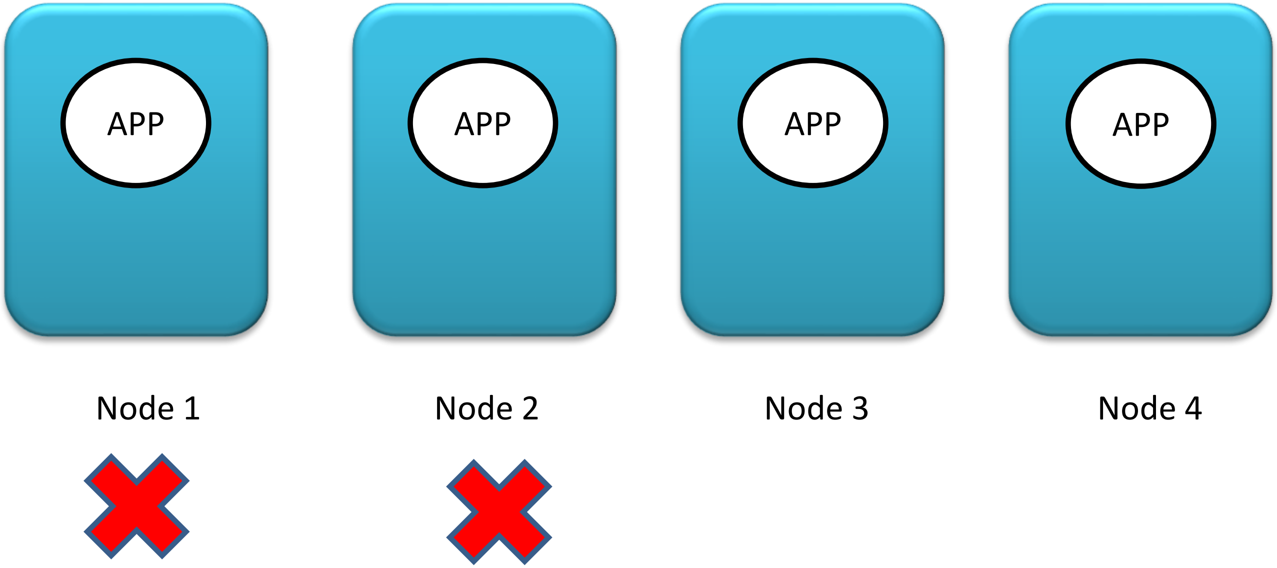
Kubernetes – Getting Started

APP Deployment – All The Way



Container Orchestration

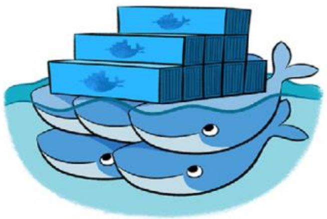


Container Orchestration

- Container Life Cycle Management
- Container deployment and start, stop, remove containers
- Scale up and scale down as per the requirement
- Container migration from one host to another in case required
- Load balancing
- Automatic Scheduling
- Monitoring

- E.g If 4 instances of container application are running on 4 nodes
- What if,
  - One of the container gets removed
  - One of the node gets down
  - Container application failed to start etc

Orchestration Solutions



Docker Swarm

kubernetes

Amazon ECS

RED HAT OPENSHIFT

Amazon EKS

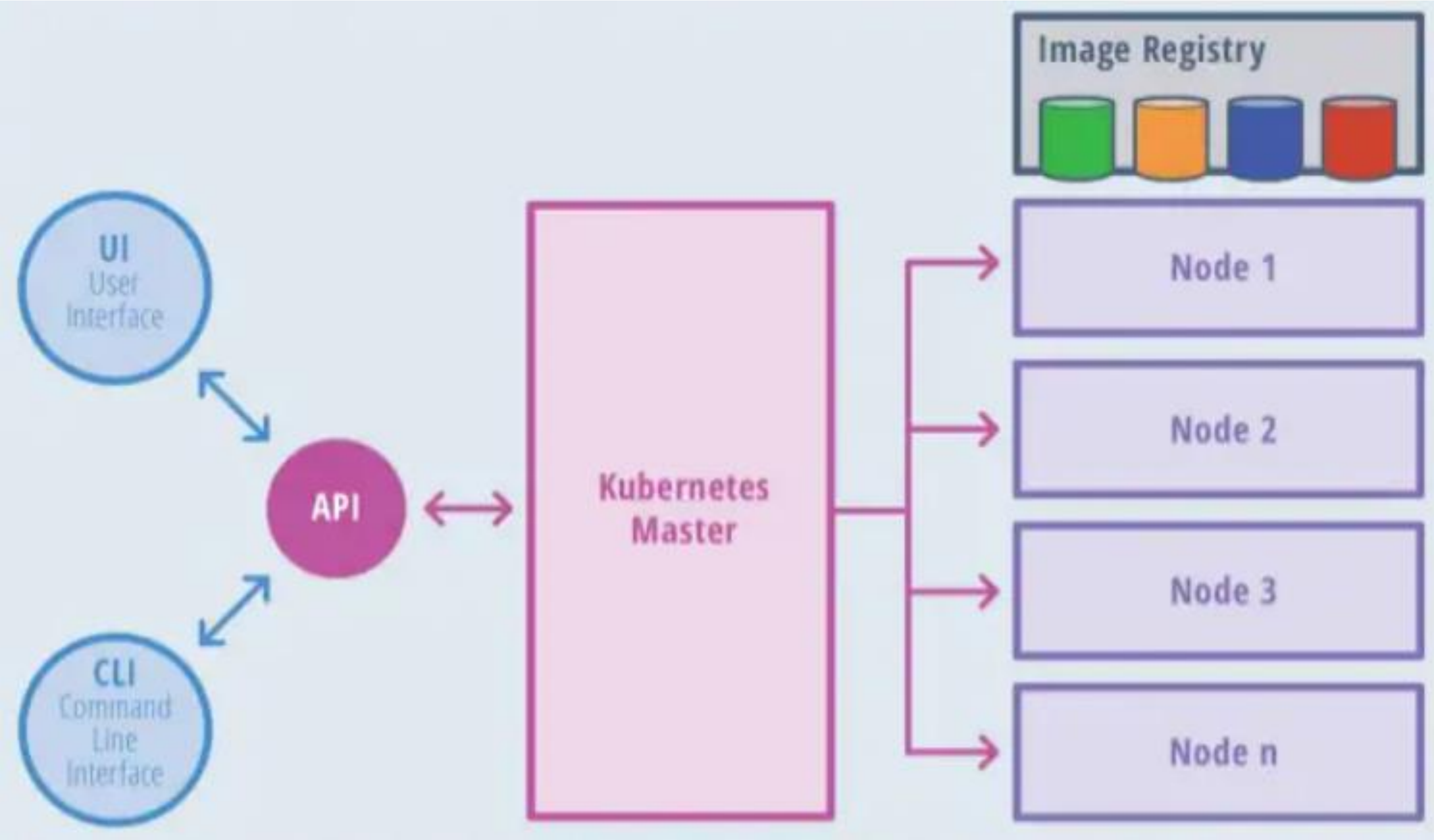
Apache MESOS

MIRANTIS

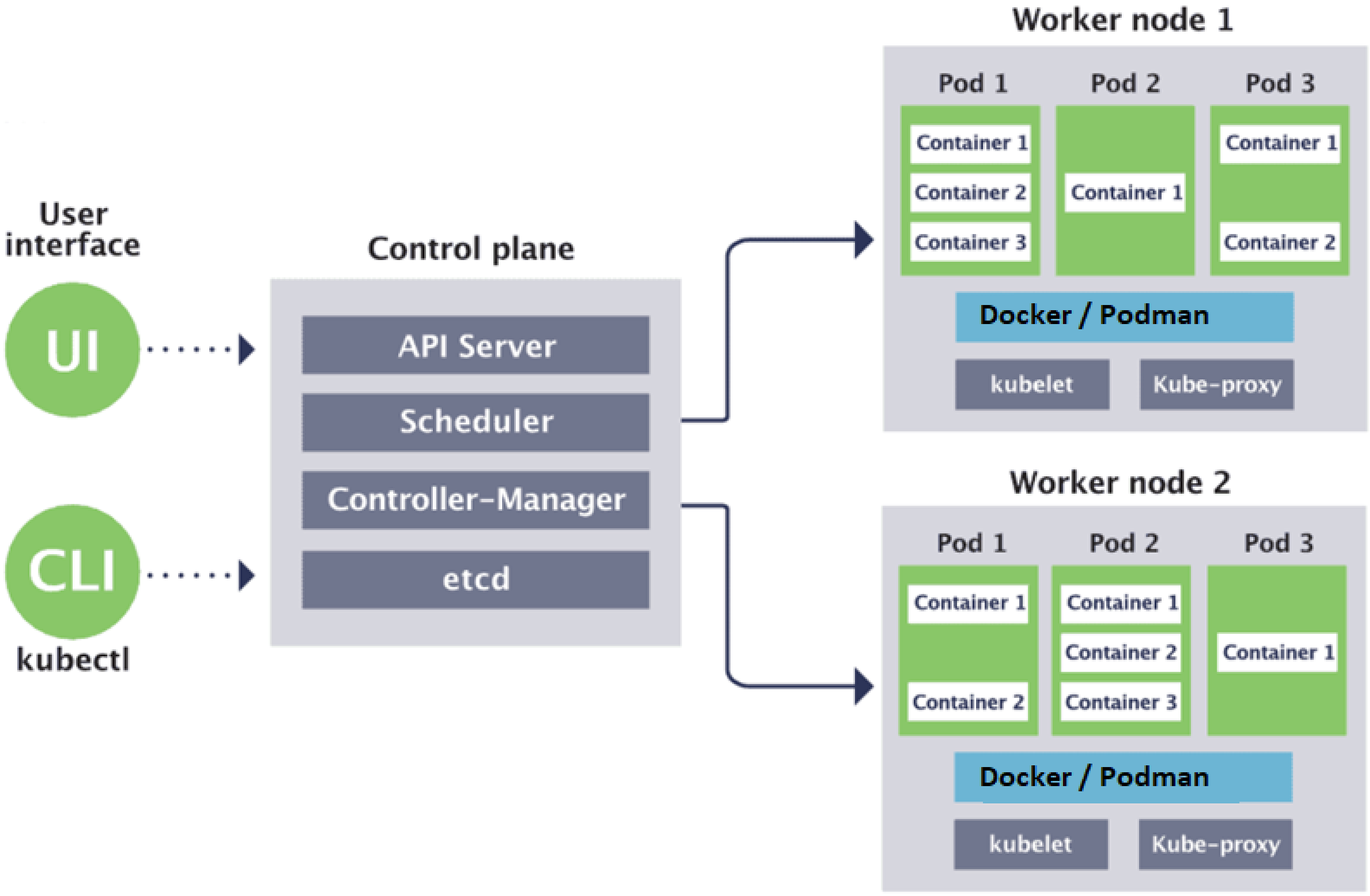
Introduction To Kubernetes

- K8s is an open-source container orchestration engine developed by Google
- It was originally designed by Google and now maintained by Cloud Native Computing Foundation (CNCF)
- K8s means – K + 8 letters between K and S + S = K8s

Simple Kubernetes Architecture



Kubernetes Architecture



Kubernetes Components

- Control Plane Components
  - Apiserver – Exposes the Kubernetes API, frontend
  - Etcd – Key value store for storing all cluster data
  - Scheduler - Scheduling of objects
  - Controller manager – Runs controller processes
- Node Components
  - Kublet – Agent makes sure container are running inside pod
  - Kube proxy – Maintain network rules and allow network communication to pods
  - Container Runtime – Responsible for running containers

Installation Methods

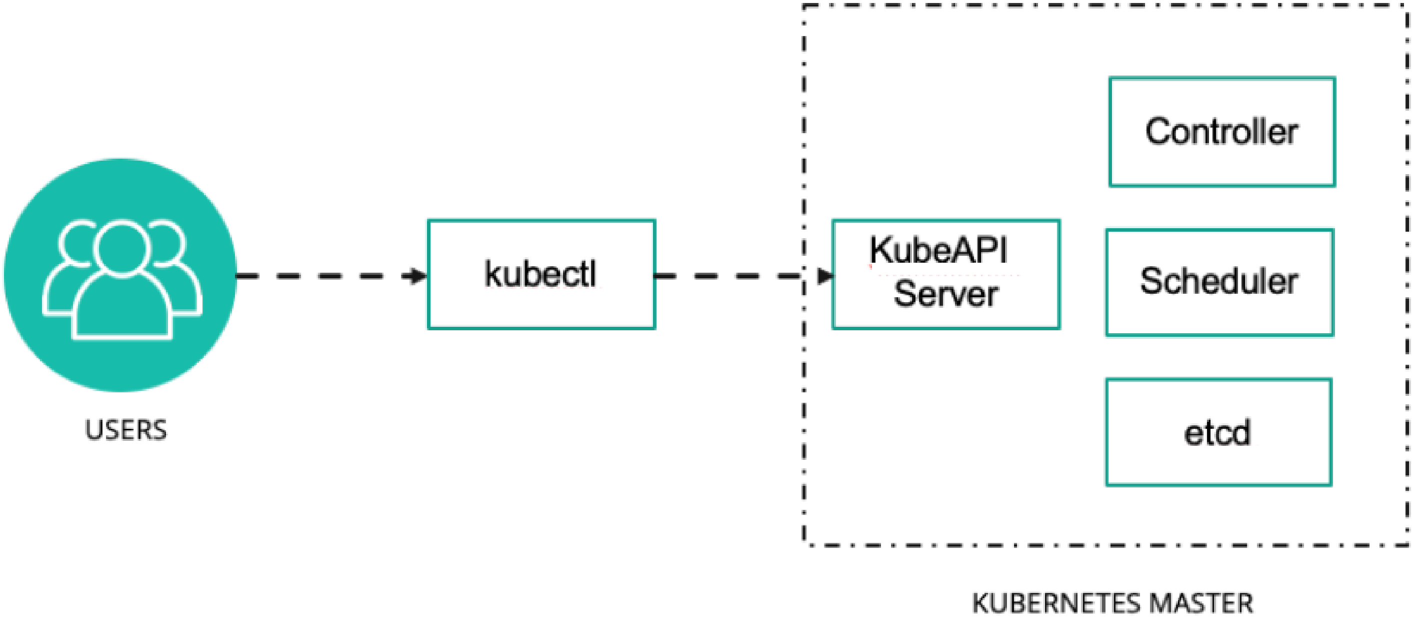
- **Minikube** – Single Node Kubernetes cluster. It is used for learning, POC, Dev environment etc.

- **Managed Kubernetes Service** – EKS, AKS, Digital Ocean etc.

- **Install manually** – Manual installation of k8s components. It is not easy and simple

Kubectl installation

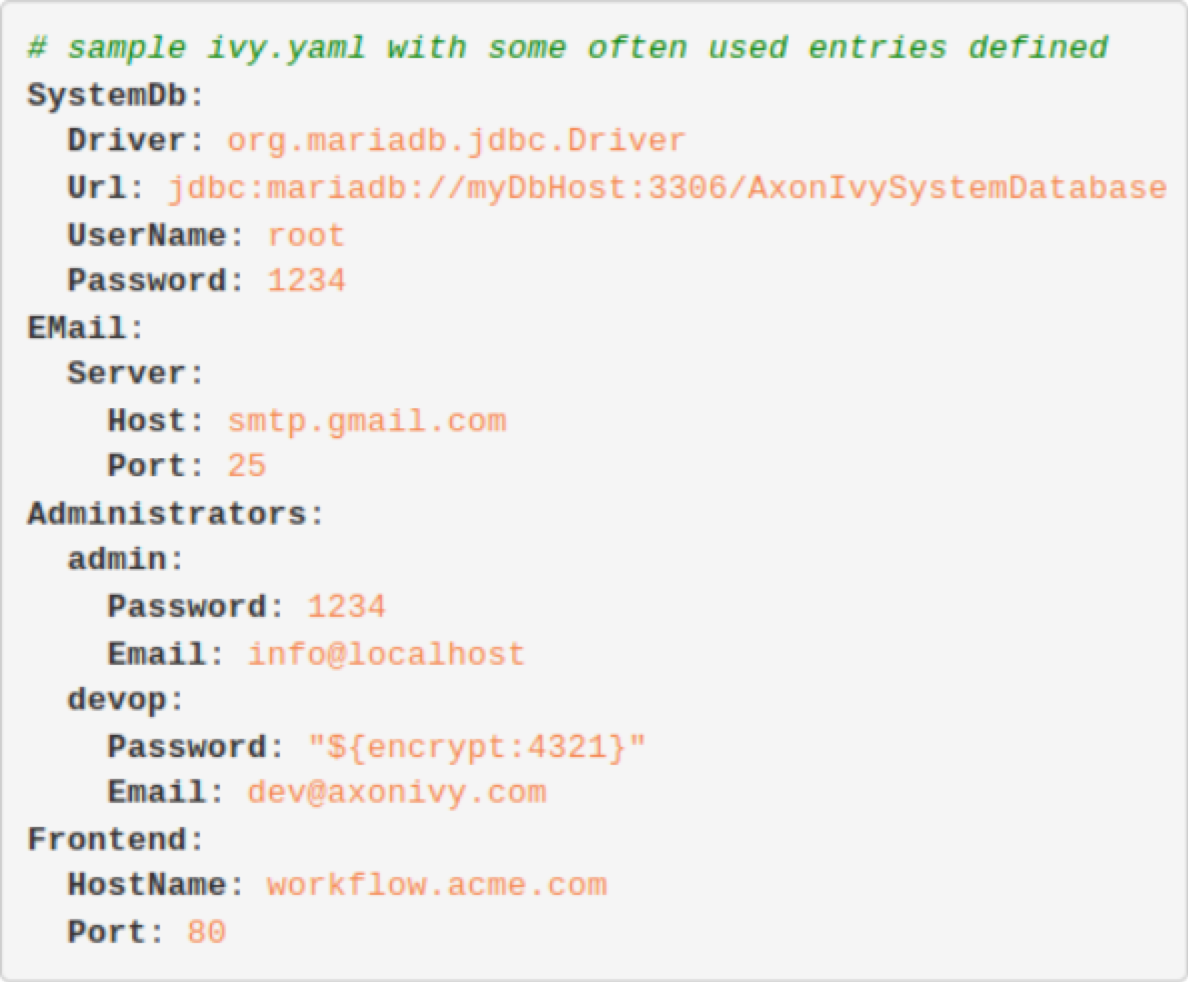
- Command line tool used to run commands against the kubernetes cluster
- You can use kubectl to deploy and manage application in cluster
- To connect from kubectl to cluster
  - ➢ DNS/IP of cluster
  - ➢ authentication token



Understanding YAML

- Human readable data serialization language
- Y- YAML
- A- Aint’s
- M – Markup
- L - Language
- Human friendly
- Widely used in programming language
- There are other types – xml, json etc.

Sample YAML



```yaml
# sample ivy.yaml with some often used entries defined
SystemDb:
  Driver: org.mariadb.jdbc.Driver
  Url: jdbc:mariadb://myDbHost:3306/AxonIvySystemDatabase
  UserName: root
  Password: 1234
EMail:
  Server:
    Host: smtp.gmail.com
    Port: 25
Administrators:
  admin:
    Password: 1234
    Email: info@localhost
  devop:
    Password: "${encrypt:4321}"
    Email: dev@axonivy.com
Frontend:
  HostName: workflow.acme.com
  Port: 80
```

Understanding K8s YAML Structure

apiVersion - Version of API e.g. V1

kind - Kind of object you create e.g. POD, deployment, service etc.

metadata - Unique name of object

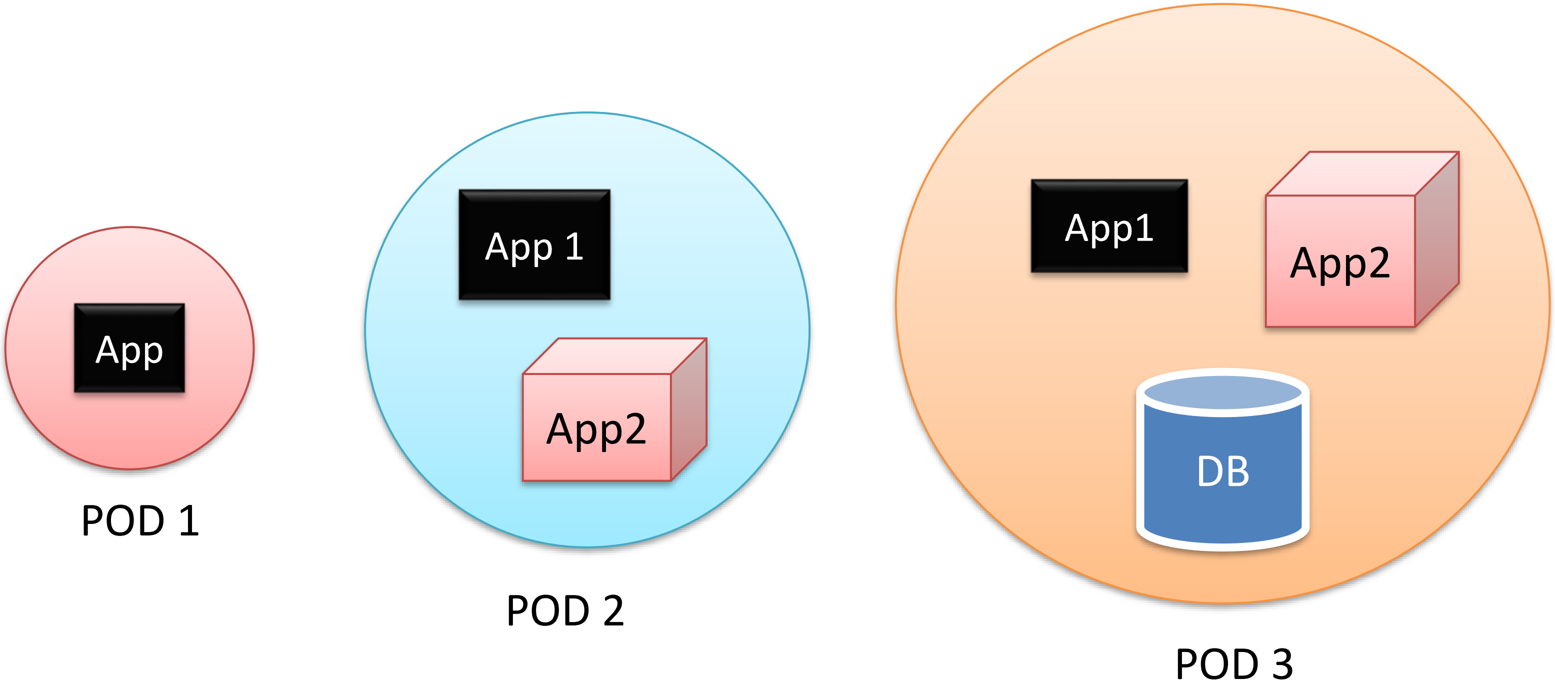
spec - Details where desired state of object. E.g. how many pods, image, cmd etc.

Sample K8s YAML File

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
  - name: mypod
    image: nginx
```

Understanding PODS

- Smallest object in K8s
- Wrapper over container
- Group of one or more containers
- All containers in POD shares resources
- Generally runs on worker node

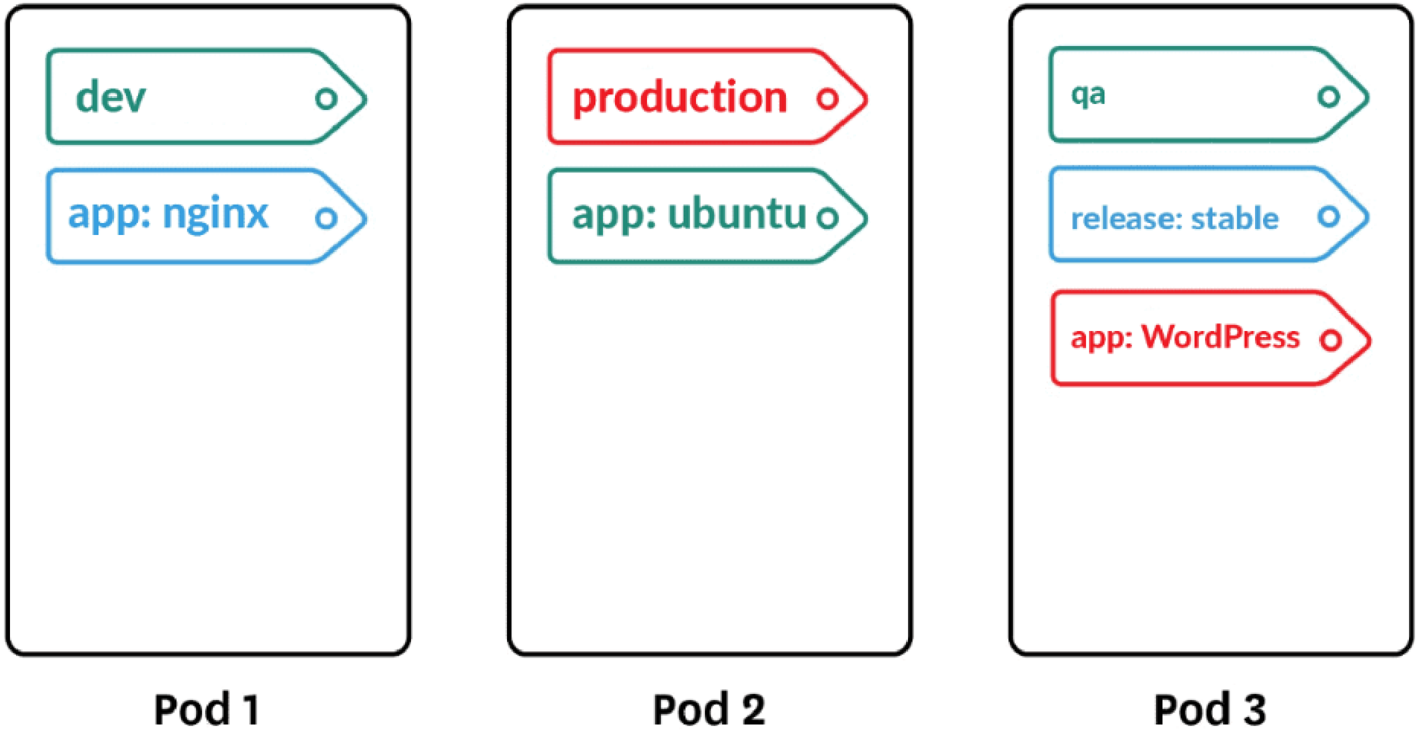


Label And Selector

- Key value pair attached to k8s objects such as pods, namespaces, deployments, service etc.
- For easily identify resource to do operation
- Helps in filtration or categorization
- Like tags in AWS
- Selectors - Used to filter objects based on label
- e.g. list all the production PODS so label as env=prod

POD Labels

- Label the PODS
- E.g. Provide the list of PODs where label as env=production


dev
app: nginx
Pod 1
production
app: ubuntu
Pod 2
qa
release: stable
app: WordPress
Pod 3