Kubernetes by Example (with Rancher UI)

Krerk Piromsopa, Ph.D.
Computer Engineering, Chulalongkorn University

Presented to NUOL iotCloudServe workshop
December 11, 2019.

CHULA ENGINEERING Foundation toward Innovation
GIST
NECTEC a member of NSTDA
UniNet
UNIVERSITI MALAYA
Asi@Connect
TEIN
European Union

Kubernetes by Examples

- Motivation
- Kubernetes Revisit
- Cluster Architecture
- Concepts
- Activities
- Stateless Deployment
- Stateful Deployment
- Pod Scaling

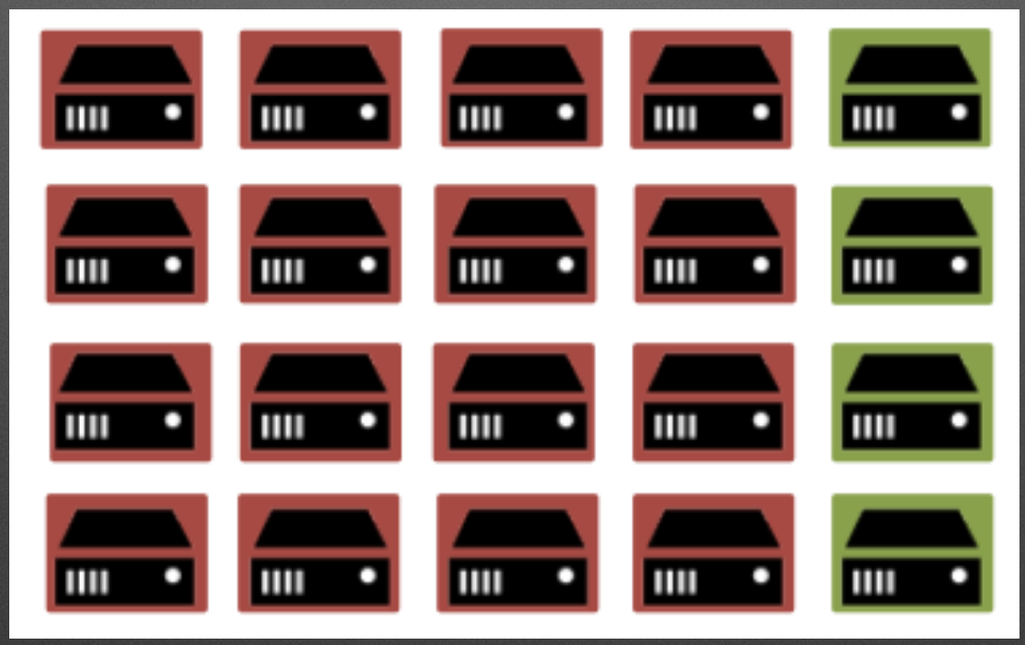
Problem with data center

Static Partitioning

Static Partitioning IS Bad

hard to utilize machines
(i.e., X GB RAM and Y CPUs)

Static Partitioning does NOT scale

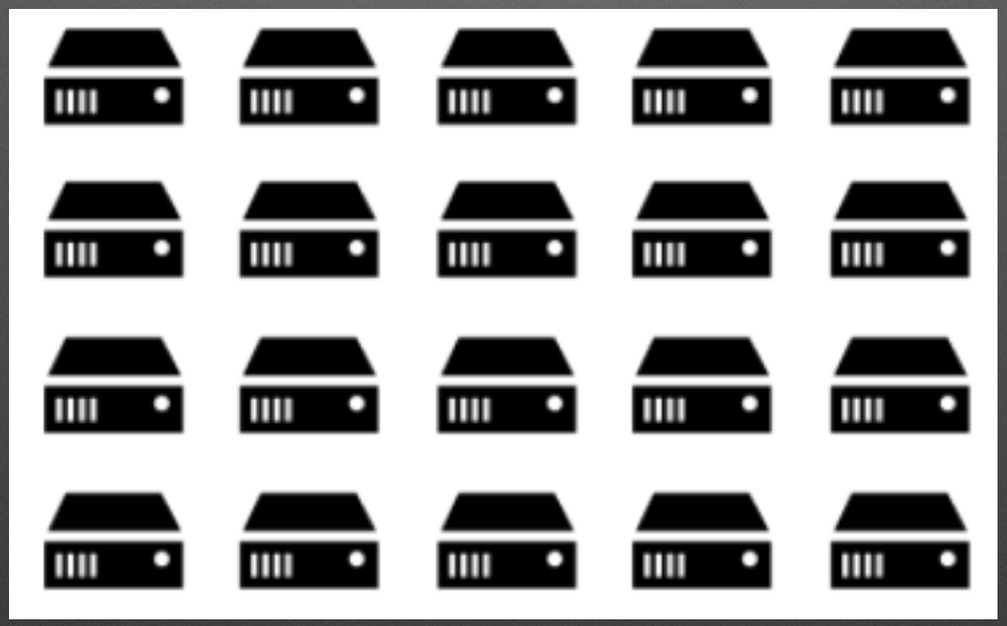


hard to scale elastically
(to take advantage of
statistical multiplexing)

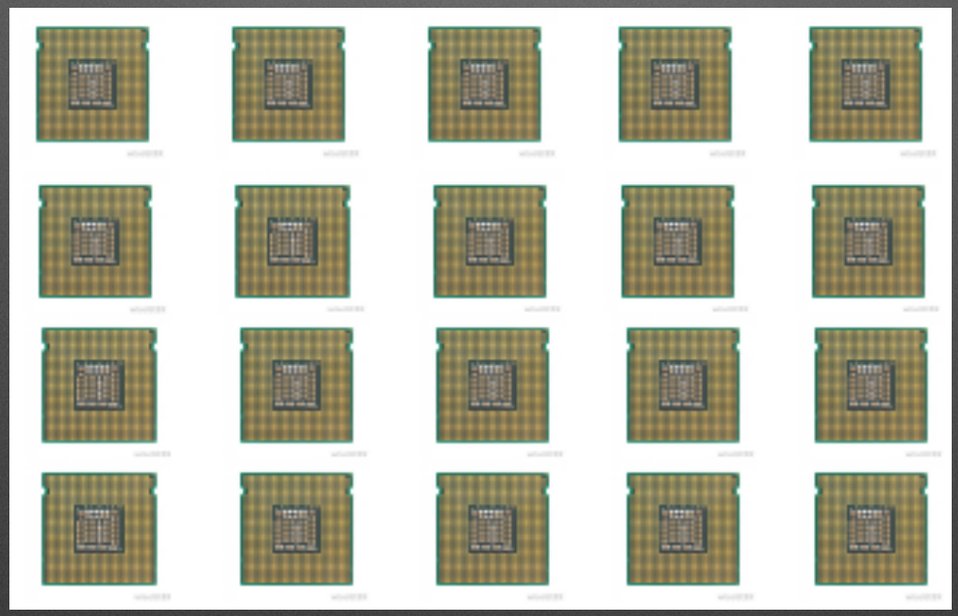
Failures === Downtime

hard to deal with failures

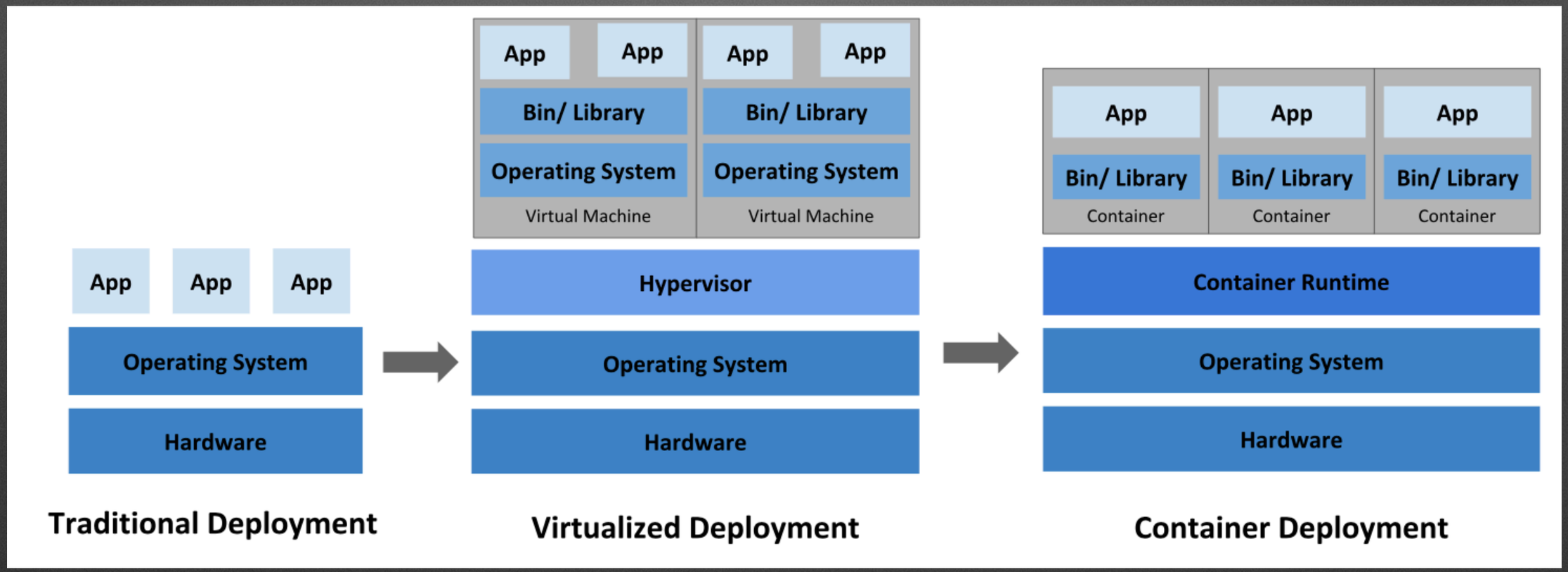
It doesn't have to be that way



Kubernetes=== Datacenter



Life before Kubernetes

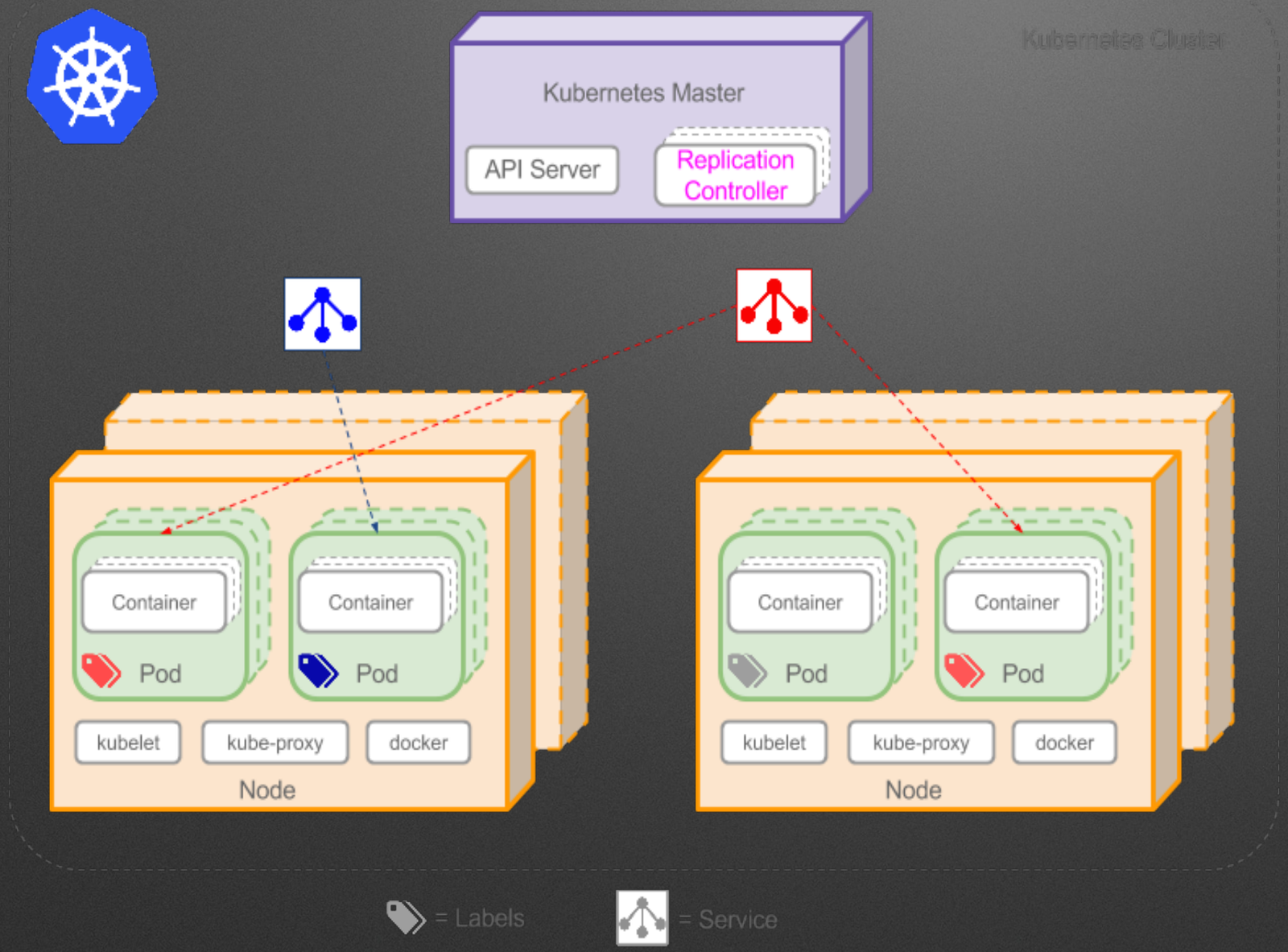


Picture from kubernetes.io

Kubernetes Concepts (Revisit)

- Cluster Architecture
- Workloads
  - Pod
    - A set of containers
  - Controllers
    - Deployments
      - ReplicaSet
    - DaemonSet
  - StatefulSets
  - CronJob
  - Job (batch)
- Services
- Ingress (Reverse Proxy)
- Storage
  - Volumes/Claims

Cluster Architecture



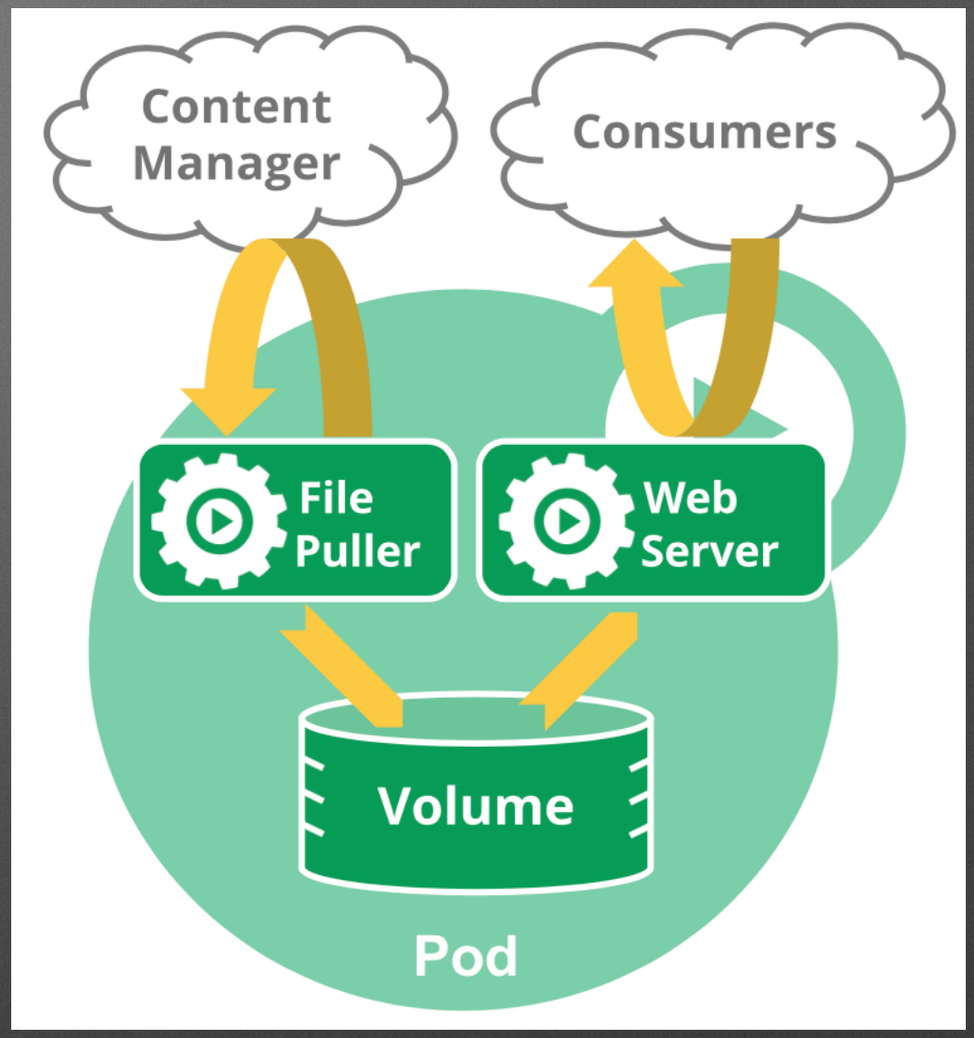
- Master
- Node/Workers
- Kubelet
- Kube-proxy
- docker

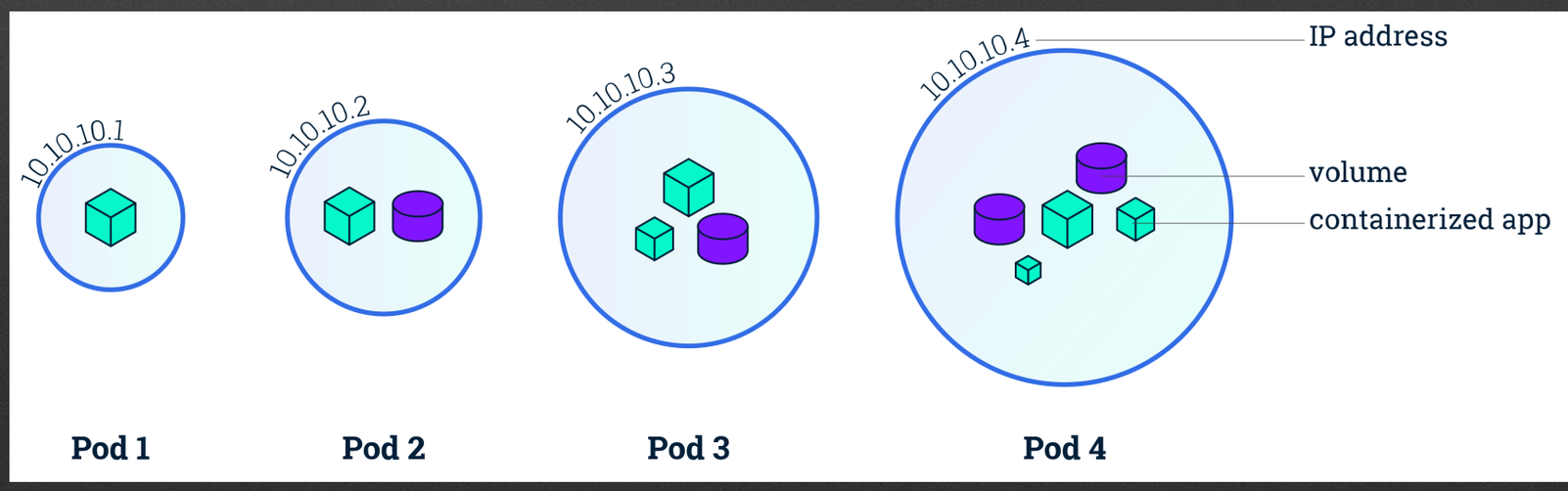
Kubernetes provides:

- Service Discovery and load balancing
- Storage Orchestration
- DevOps Operation
  - Rolling update
- Self Healing
- Secret and Configuration Management
- scheduling (container placement)

Pods

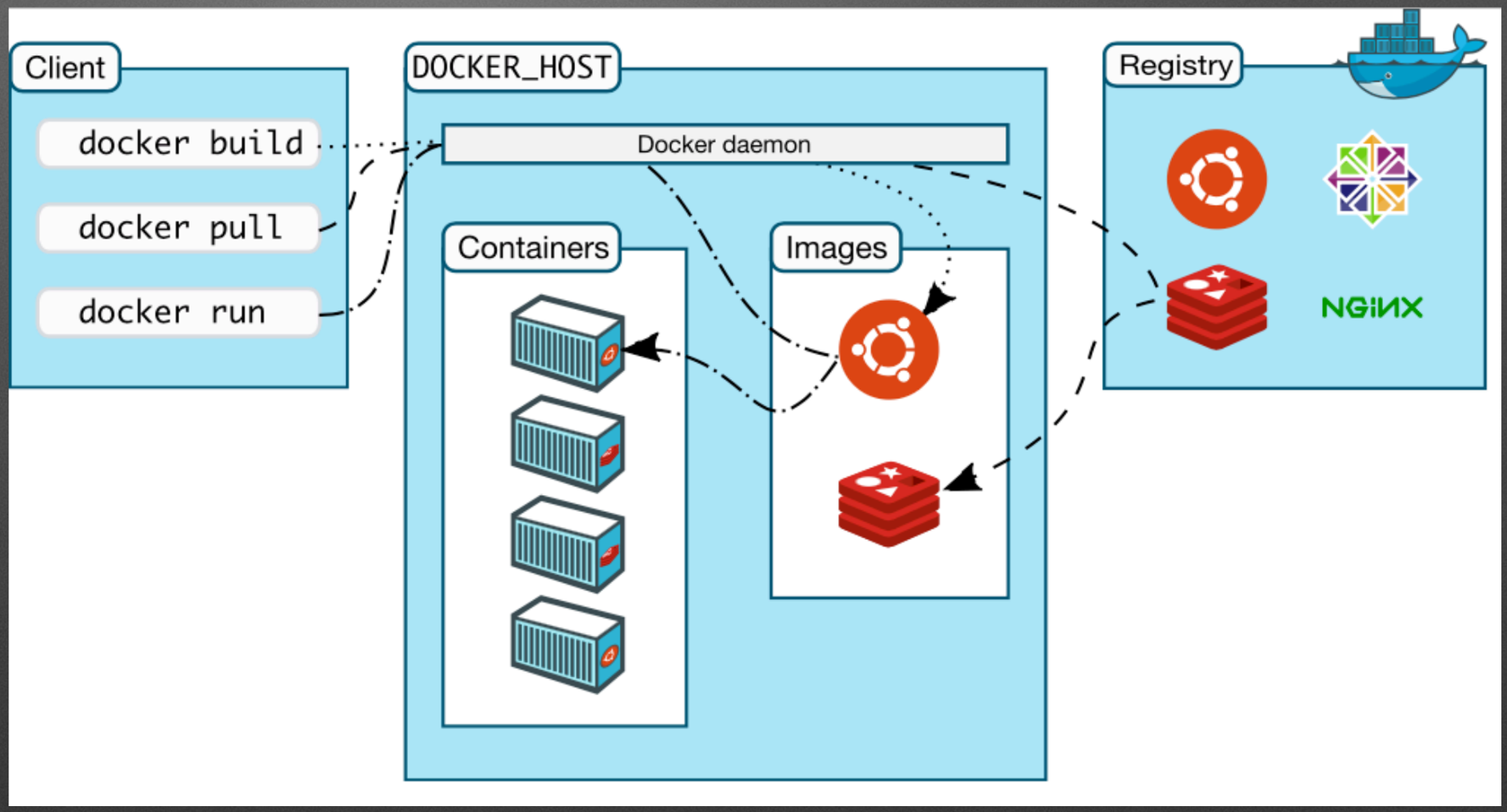
- A pod is a single set of containers.
- A container is an instance of image from registry.
- One IP per pod (think of it as a machine)
- Pod
  - App Container(s)
  - Sidecar Container
  - Init Container
  - Adapter





Containers

- Same as docker containers.
- An instance of predefined image pulling from registry.

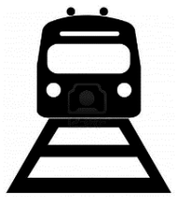


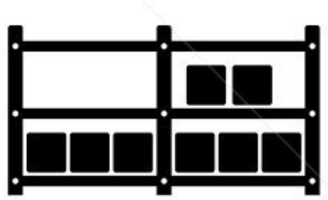
Cargo Transport Pre-1960

Multiplicity of Goods

Do I worry about how goods interact (e.g. coffee beans next to spices)

Multipilicity of methods for transporting/storing

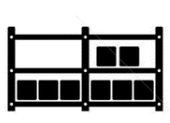




Can I transport quickly and smoothly (e.g. from boat to train to truck)

Picture from docker public presentation

Also a matrix from hell

	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
							

Picture from docker public presentation

Solution: Intermodal Shipping Container

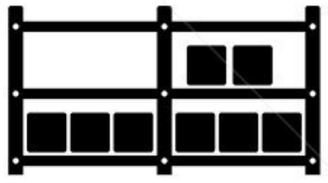
Multiplicity of Goods

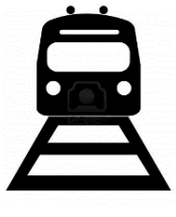
Do I worry about how goods interact (e.g. coffee beans next to spices)

A standard container that is loaded with virtually any goods, and stays sealed until it reaches final delivery.

…in between, can be loaded and unloaded, stacked, transported efficiently over long distances, and transferred from one mode of transport to another

Multiplicity of methods for transporting/ storing





Can I transport quickly and smoothly (e.g. from boat to train to truck)

Picture from docker public presentation

Docker is a shipping container system for code

Multiplicity of Stacks

Static website | User DB | Web frontend | Queue | Analytics DB

Do services and apps interact appropriately?

An engine that enables any payload to be encapsulated as a lightweight, portable, self-sufficient container…

…that can be manipulated using standard operations and run consistently on virtually any hardware platform

Multiplicity of hardware environments

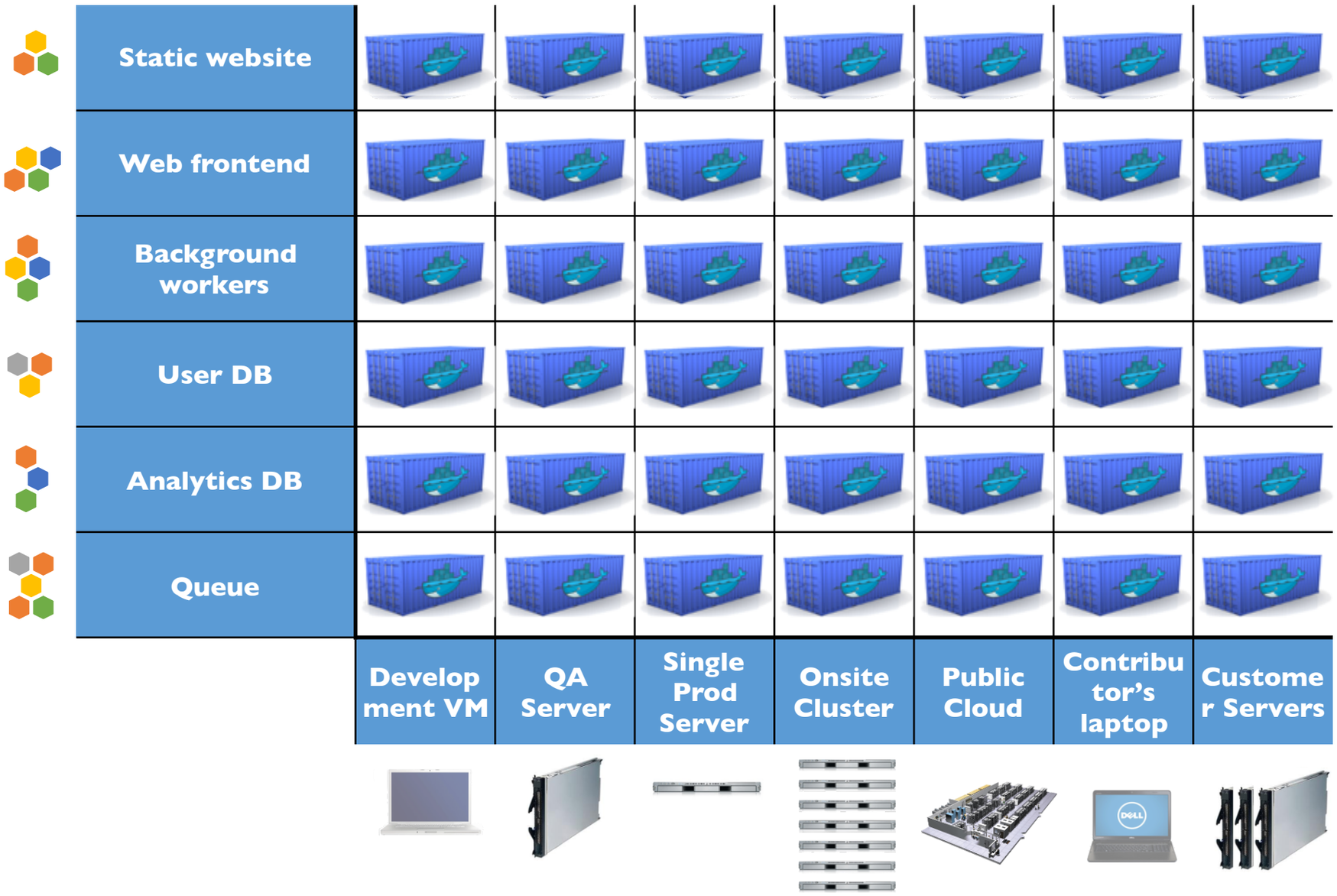
Can I migrate smoothly and quickly

docker

Developme nt VM | QA server | Customer Data Center | Public Cloud | Production Cluster | Contributor's laptop

Picture from docker public presentation

Docker eliminates the matrix from Hell

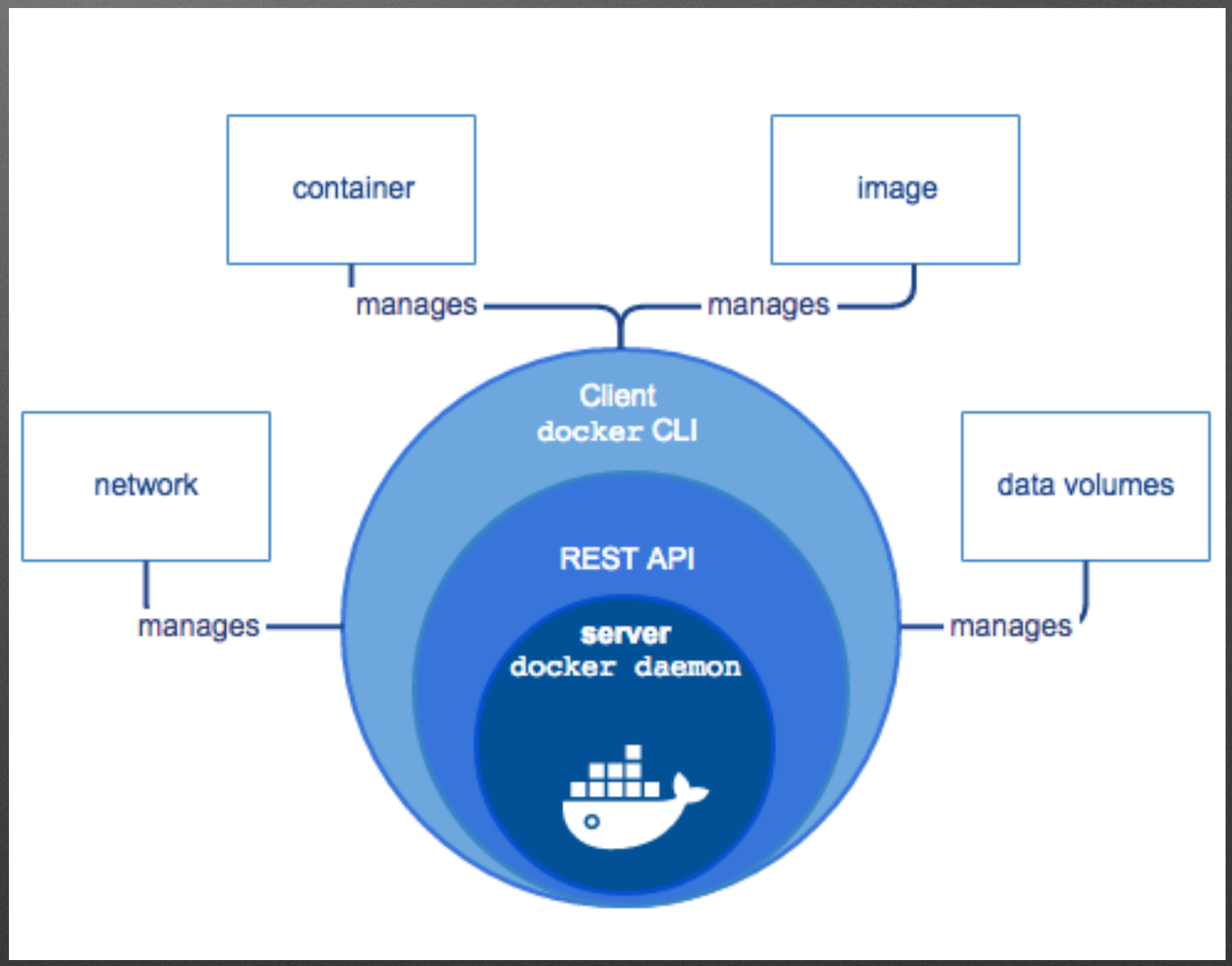


docker

Picture from docker public presentation

Containers

- Docker/Container Abstraction
- Storage
- Network
- Management (start/stop)



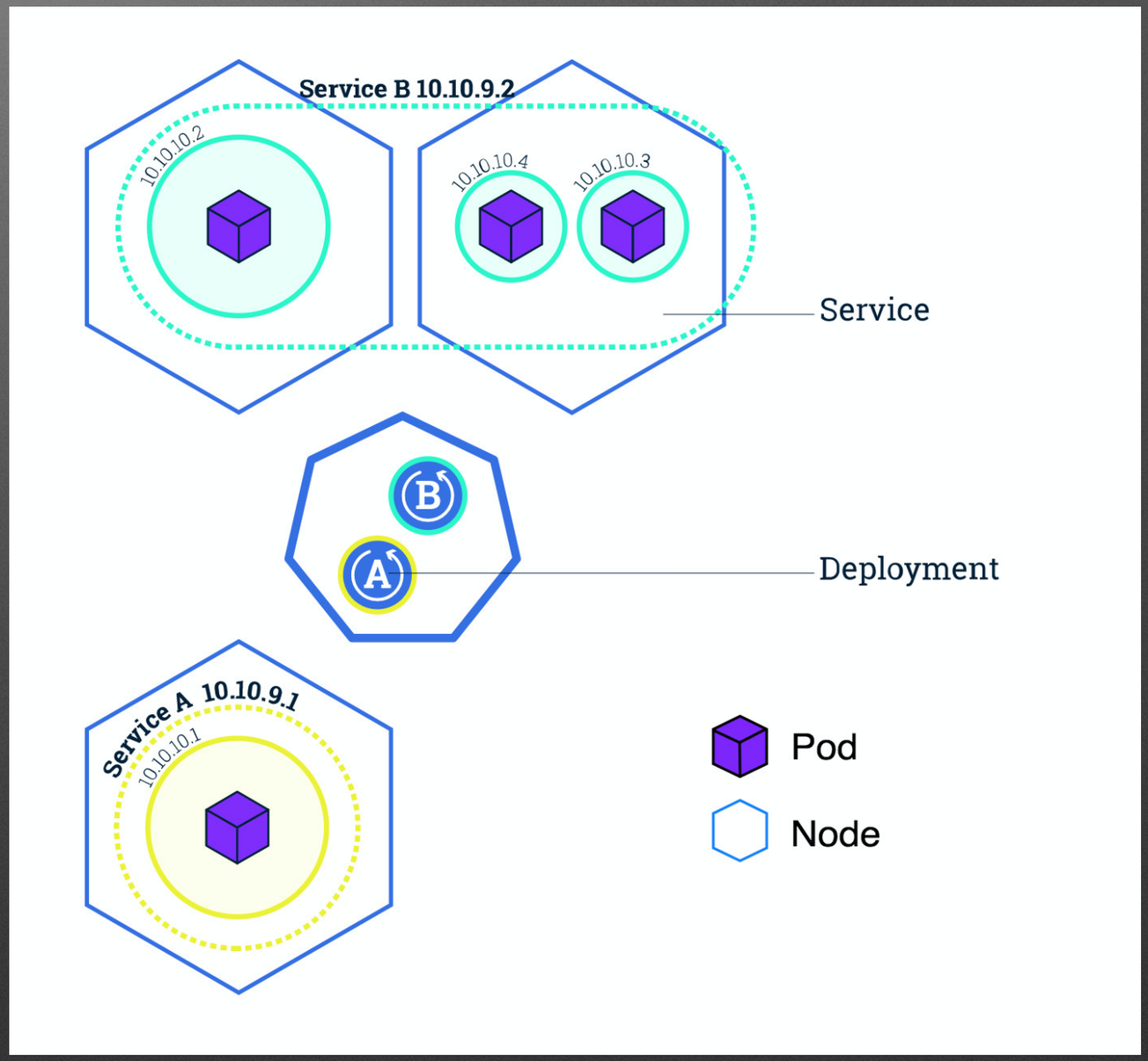
Controllers

- Deployments
  - ReplicaSet + Pod
- StatefulSet
  - Deployments with naming scheme (e.g. pod-0, pod-1, ….)
- DaemonSet
  - One pod per node
- Job - run until complete
- CronJob - Scheduled Job
  (similar to cron on Linux/Unix)

Services

Group deployments together as a unit of service

- Expose a deployment to the network for others to use.
- Headless Service - use internally in the cluster
- Standard Service - associated with a cluster IP or nodes (port map)



Ingress/Load Balancing

Smart uses of resources (network)

- Manage external access to services (HTTP, HTTPS)
- 2 features
  - Share IPs with several host name (named virtualhost)
    - Map several services to different paths of a host
      - www.example.com (10.0.0.1) - sample-service
        - /app1/ - to app1-service
        - /app2/ - to app2-service
      - www.test.com (10.0.0.1) - test-service

Storage

- Volume - a storage drive
  - attached to a pod,
    (deleted when the pod is deleted)
- Persistent Volumes
  - Persist without POD
  - Static - local disk, network fs
  - Dynamic - provisioning on demand

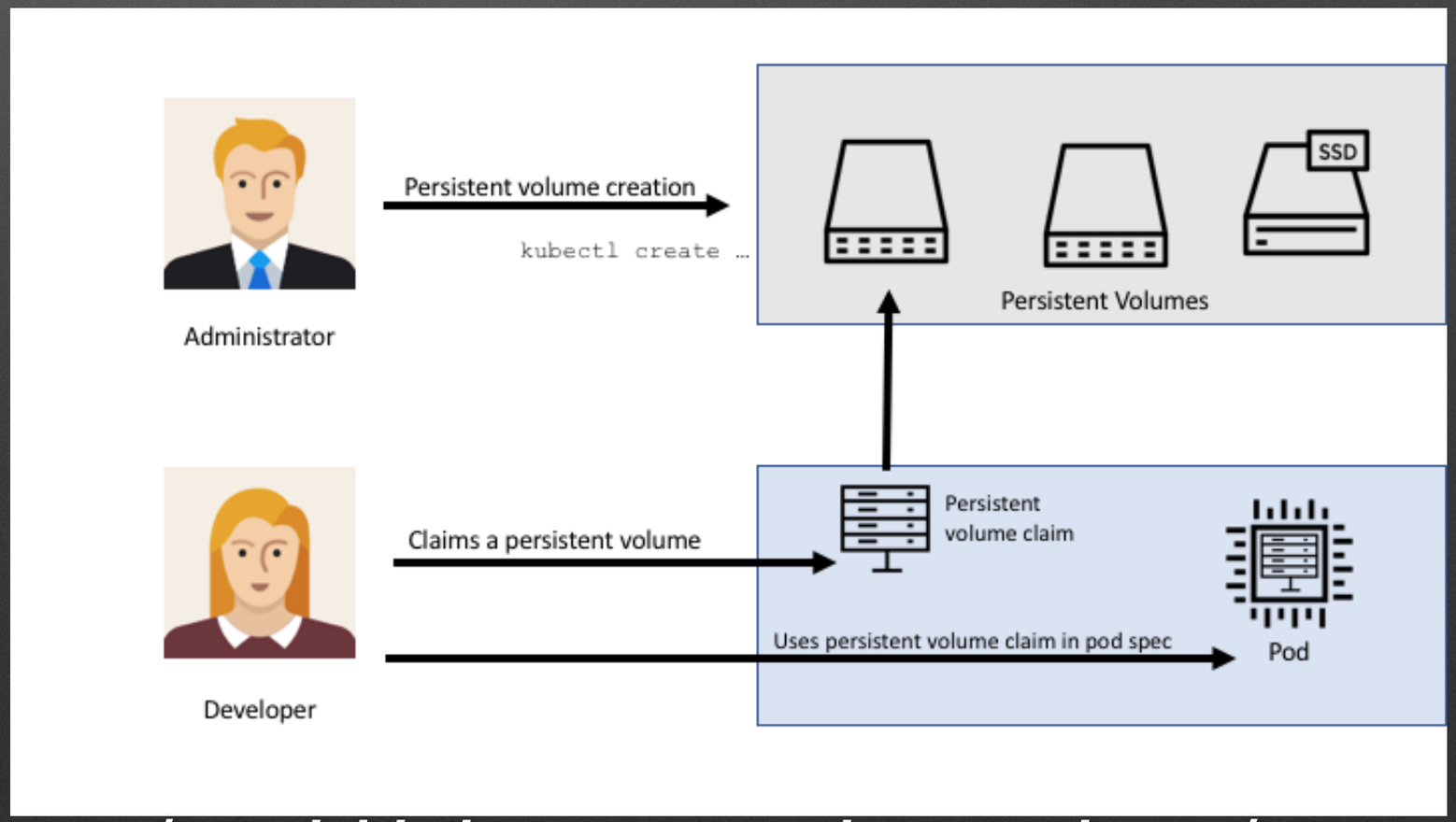
Types of Volumes

Kubernetes supports several types of Volumes:

- awsElasticBlockStore
- azureDisk
- azureFile
- cephfs
- cinder
- configMap
- csi
- downwardAPI
- emptyDir
- fc (fibre channel)
- flexVolume
- flocker
- gcePersistentDisk
- gitRepo (deprecated)
- glusterfs
- hostPath
- iscsi
- local
- nfs
- persistentVolumeClaim
- projected
- portworxVolume
- quobyte
- rbd
- scaleIO
- secret
- storageos
- vsphereVolume

Dynamic Volumes

- Dynamically claim (using Persistent Volume Claim) from storage class (a lot for allocation units)
- Pods view the claims as volumes.



Picture from https://portworx.com/tutorial-kubernetes-persistent-volumes/

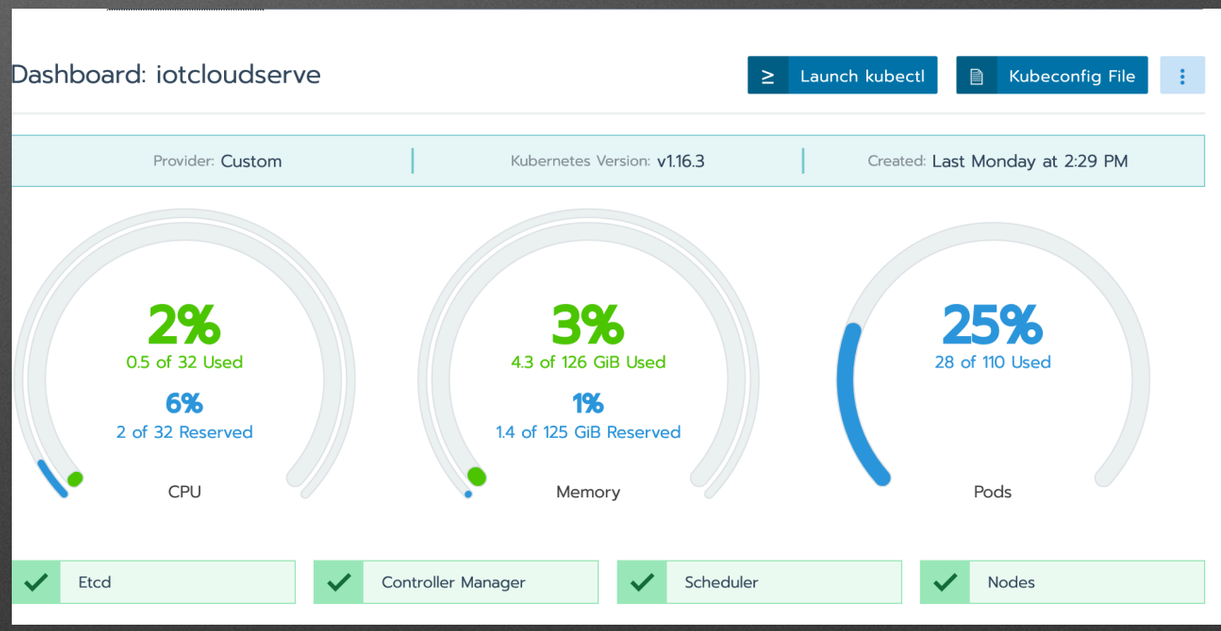
Horizontal Pod Autoscaling

- Monitor the resources of the POD
- Scaling according to rules (CPU, Memory)

Rancher

- An open-source CNCF certified kubernetes distribution. Jointly developed by RedHat and Rancher Labs.

- Easy GUI for Kubernetes.



Let’s try it.