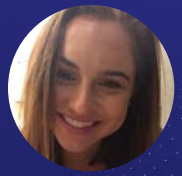
Kubernetes for the Virtualization Admin



Julia Furst Morgado
Global Technologist
Veeam® Software
@juliafmorgado

Kubernetes for the virtualization admin

Kubernetes is just another platform concentrating on containerization. You are all familiar with virtualization and most of you will be focused on VMware vSphere which as you know is a platform for virtualization and more recently a place in which you can also now run Kubernetes in the form of Tanzu Kubernetes Grid.

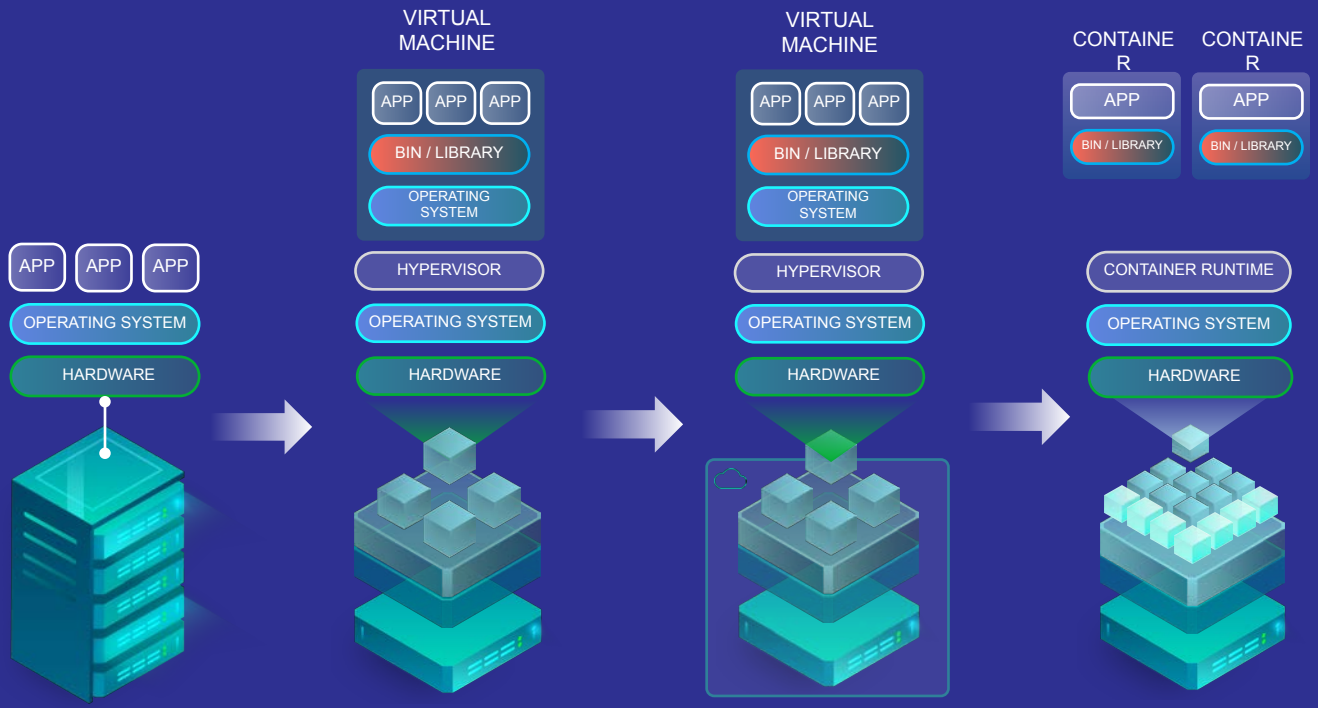
In this session, we will walk through the correlation between virtualization and Kubernetes and how it is not that far away from what you are used to with vSphere. At the end of the day, Kubernetes is made up of compute, storage and networking — very much the same as a virtualization platform. We are going to breakdown some of the seemingly daunting aspects of Kubernetes and also give you the building blocks of Kubernetes.

Agenda

- Intro to containers
- Intro to Kubernetes
- Kubernetes on vSphere
- Kubernetes networking with vSphere
- Kubernetes storage with vSphere
- vSphere with Tanzu
- Day two operations

Intro to containers

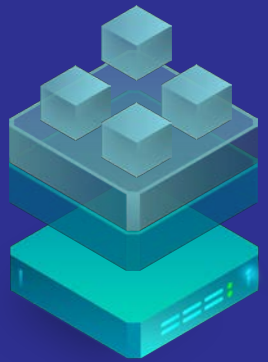
The evolution of infrastructure



None of these have gone away!

Physical systems

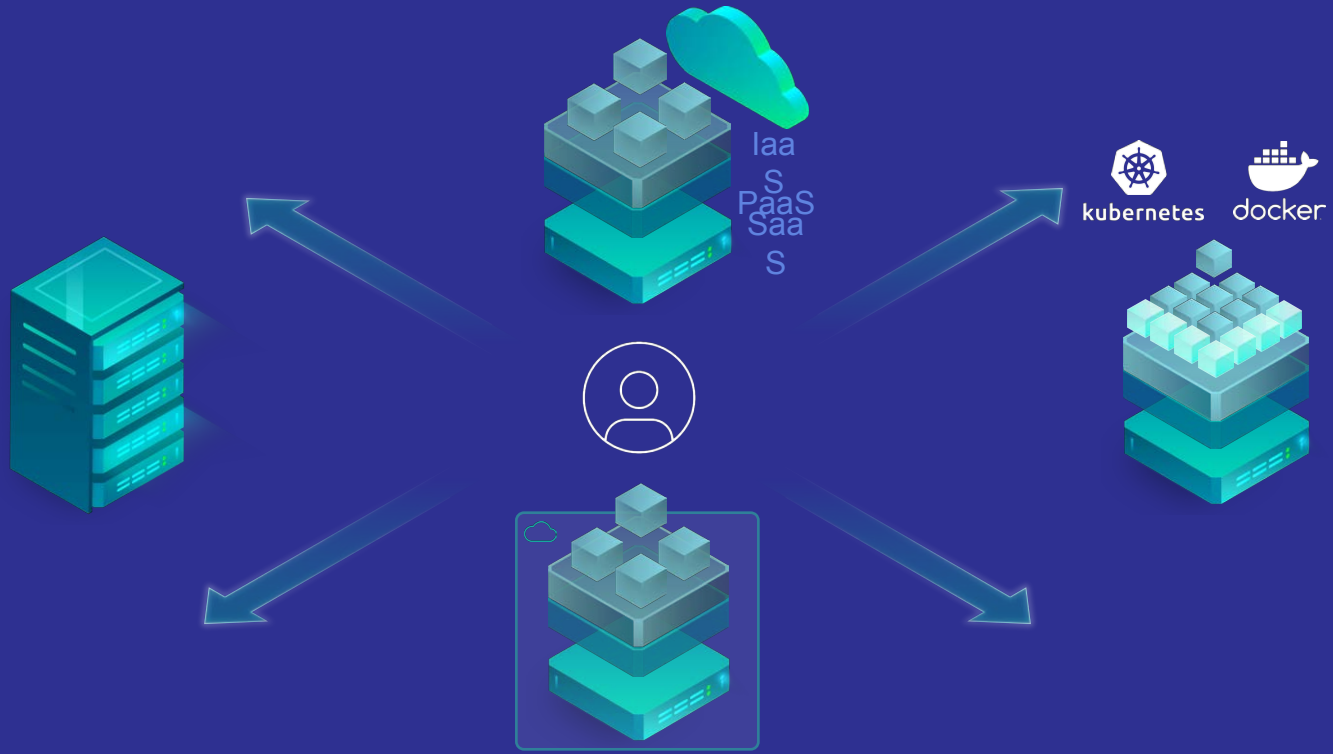
Virtualization



Cloud

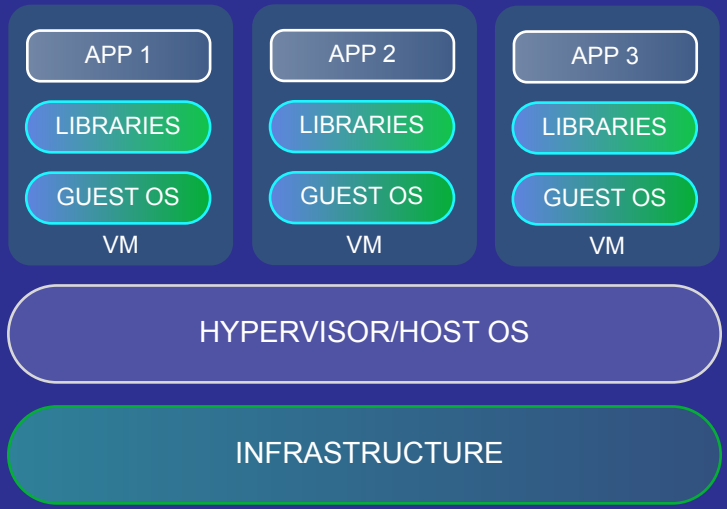
Containers

Which one do we choose?

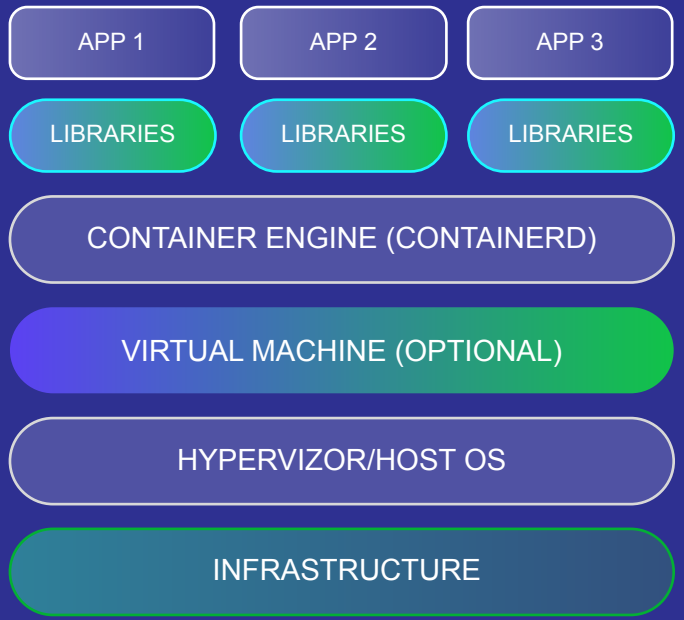


Containers ~~vs~~ & VMs

Virtualized infrastructure



Containerized infrastructure



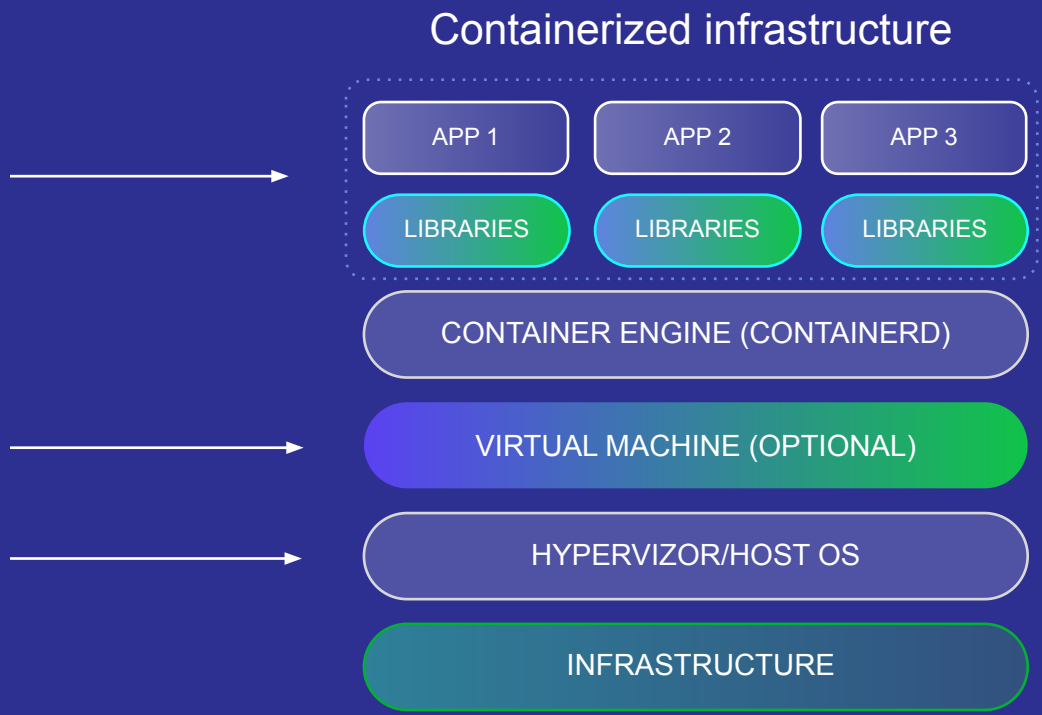
What are containers & containerization?

Just as the use of shipping containers transformed global logistics, the rise of the software container is transforming **software development and deployment**.

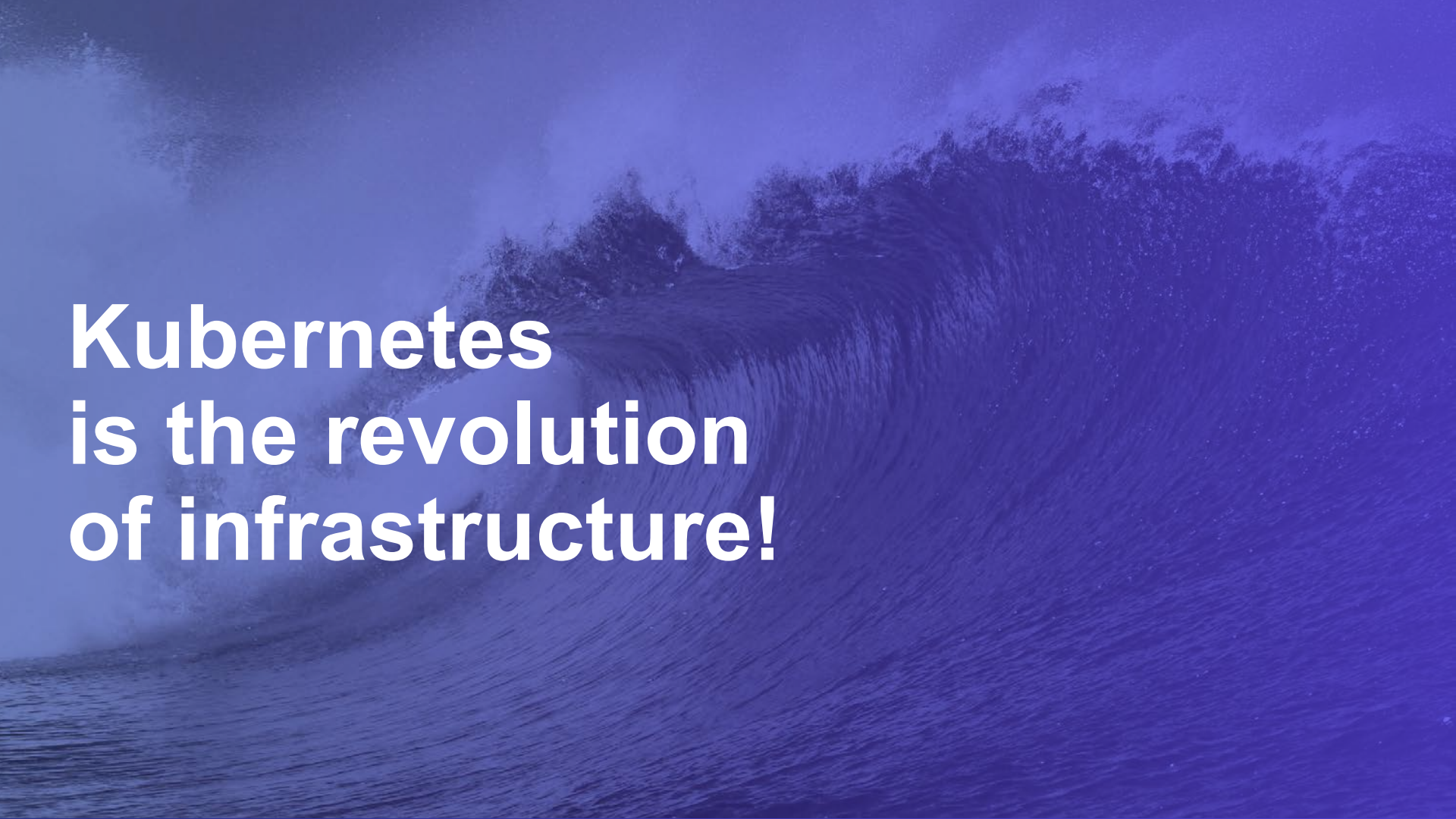
What are containers & containerization?

Remember this box as a **deployment** for later

Remember these as **nodes** for later



Intro to Kubernetes

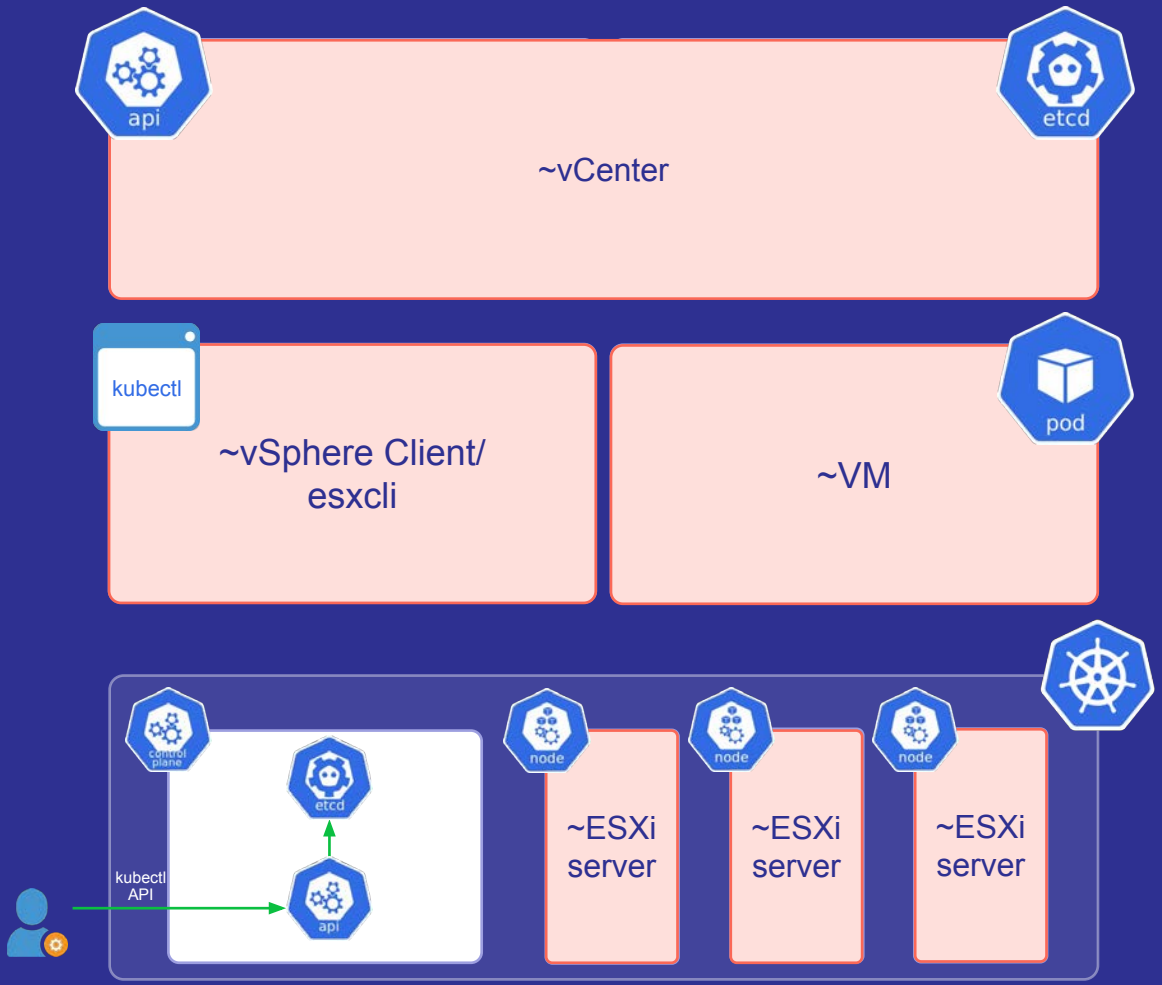

Kubernetes
is the revolution
of infrastructure!

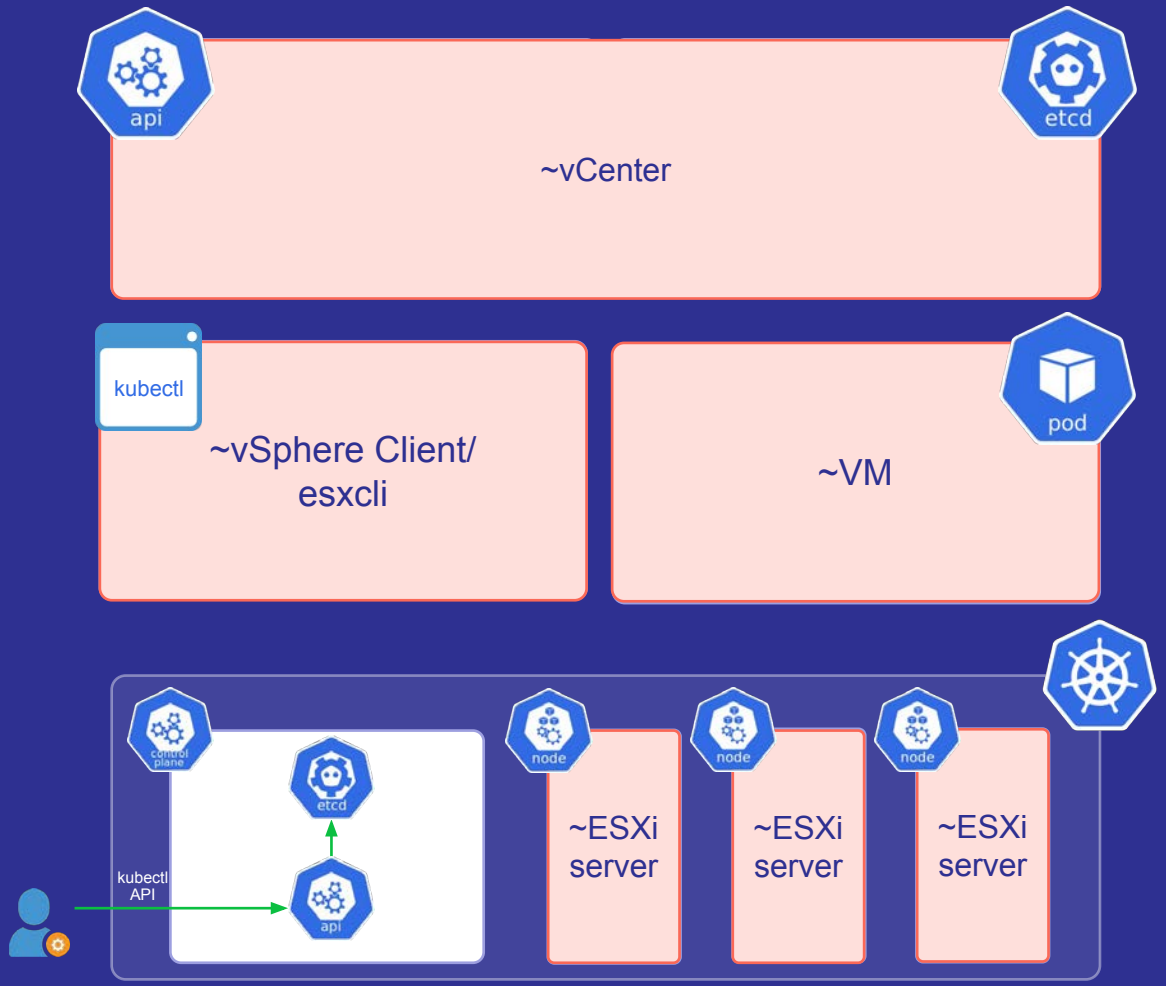
kube-apiserver
Front end of control plane

etcd
Key value store for cluster data

CLI for interacting with K8s cluster

Building block of K8s
pod ~= container





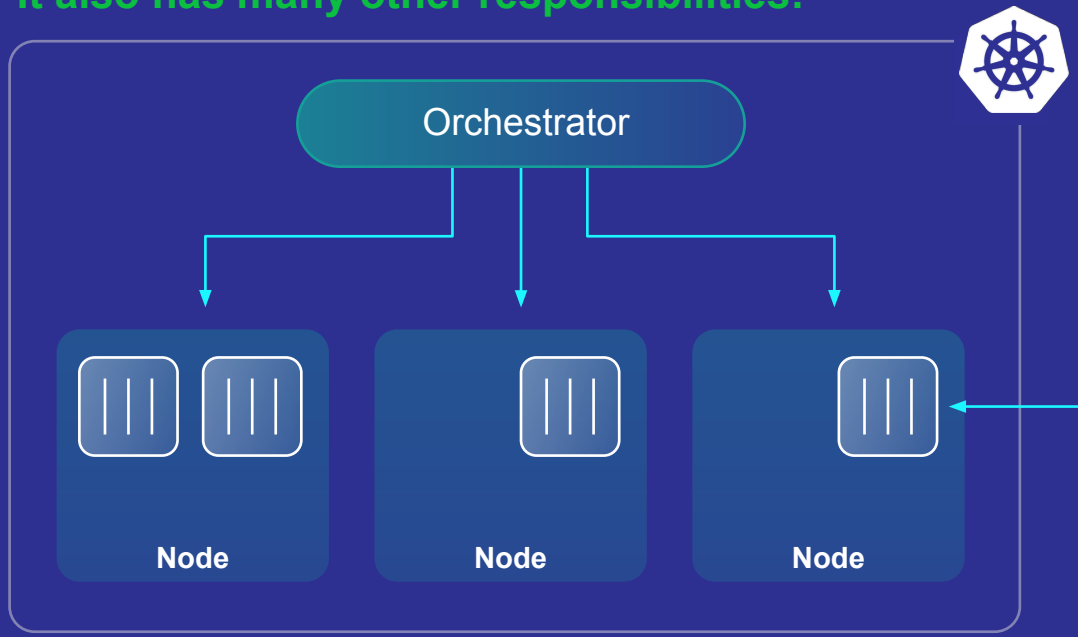
Kubernetes = container orchestration

Container orchestration manages the deployment, placement and lifecycle of containers.
It also has many other responsibilities:

- Cluster management federates hosts into one target.
- Schedule management distributes containers across nodes through the scheduler.
- Service discovery knows where containers are located and distributes client requests across them.
- Replication ensures that the right number of nodes and containers are available for the requested workload.
- Health management detects and replaces unhealthy containers and nodes.

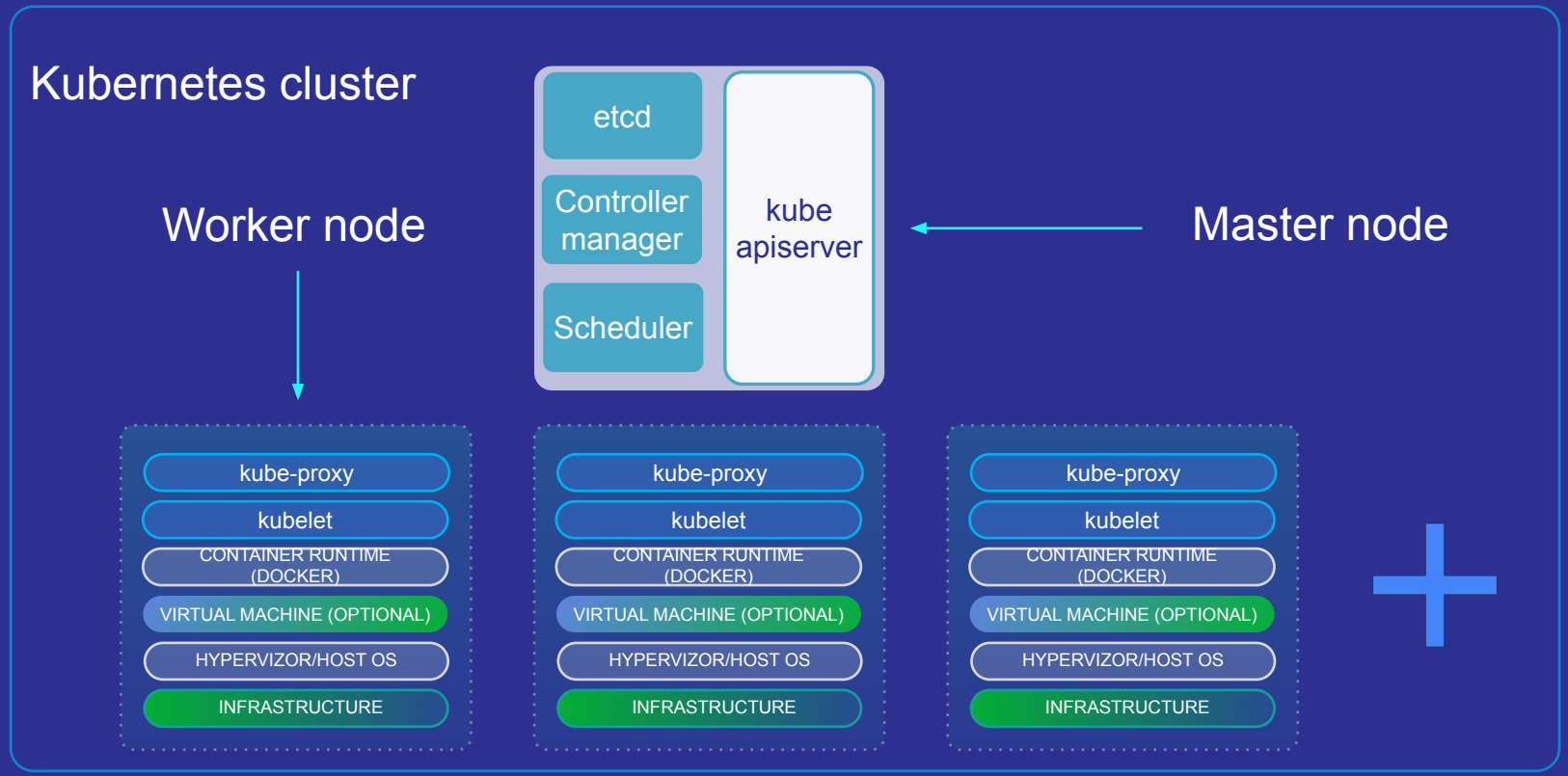
Kubernetes = container orchestration

Container orchestration manages the deployment, placement and lifecycle of containers.
It also has many other responsibilities:



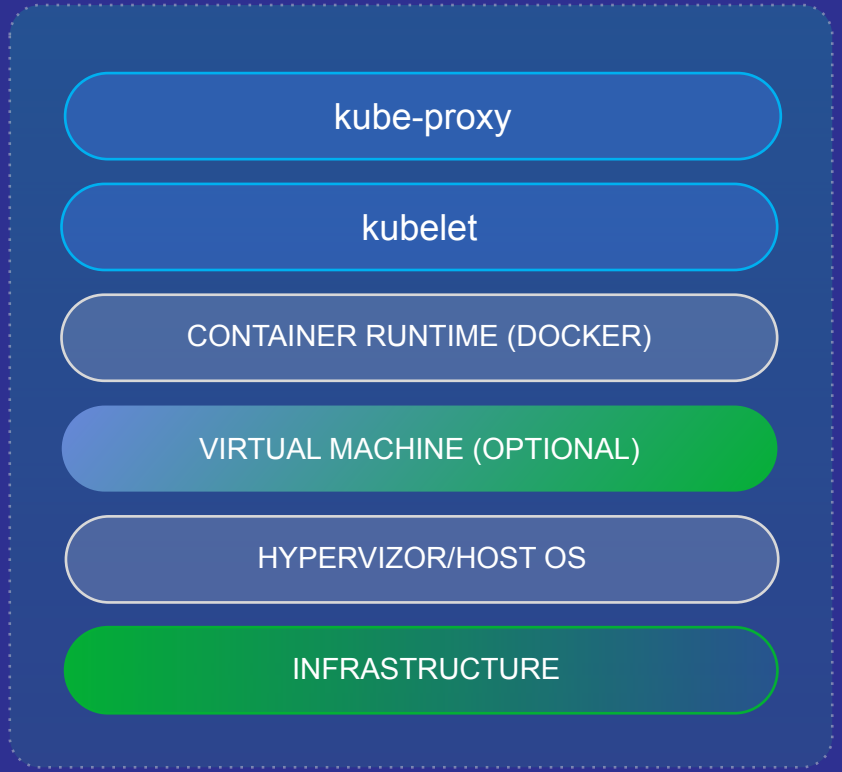
These are **containers** being orchestrated

Cluster architecture

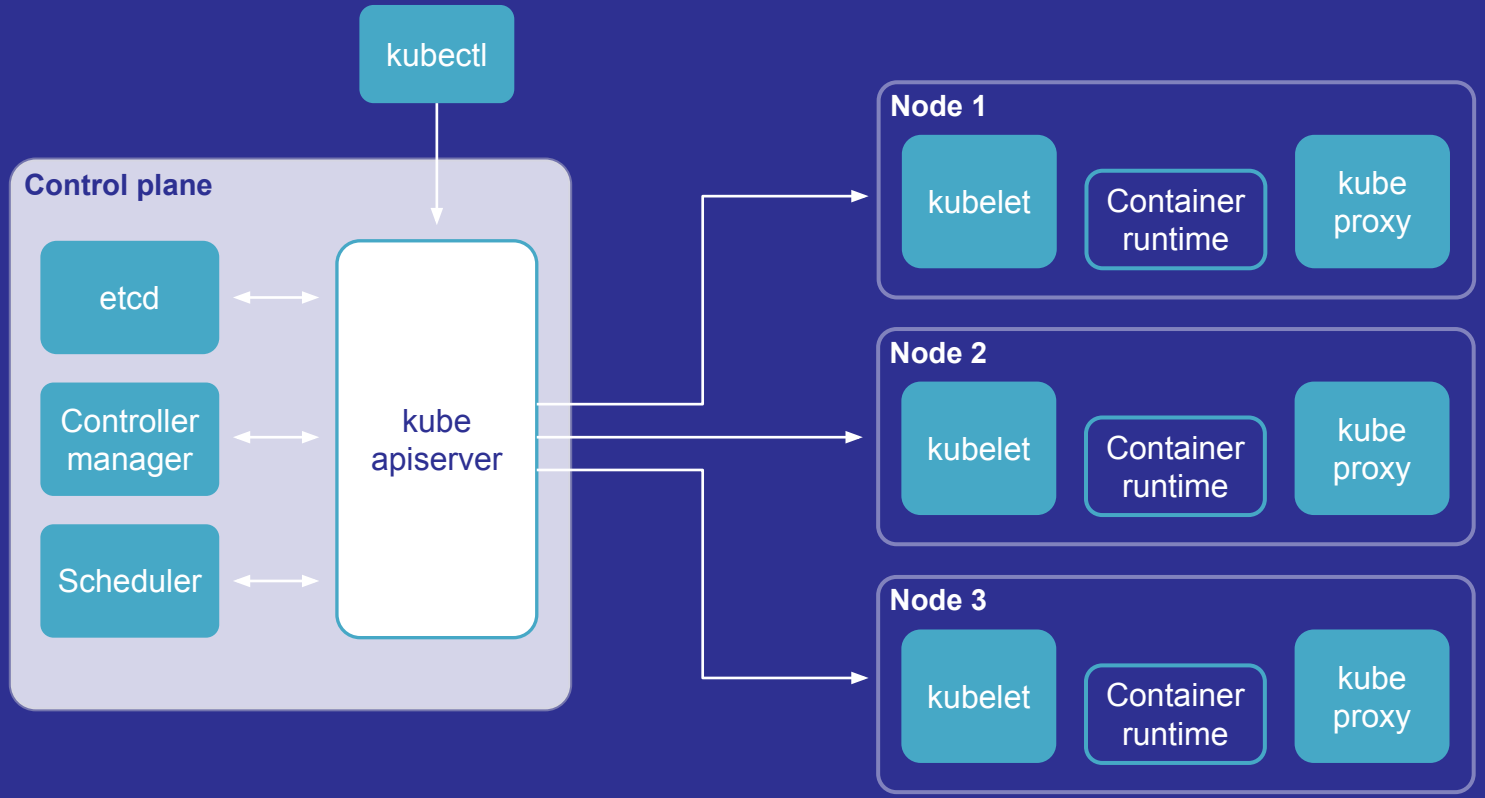


The building blocks – **worker node**

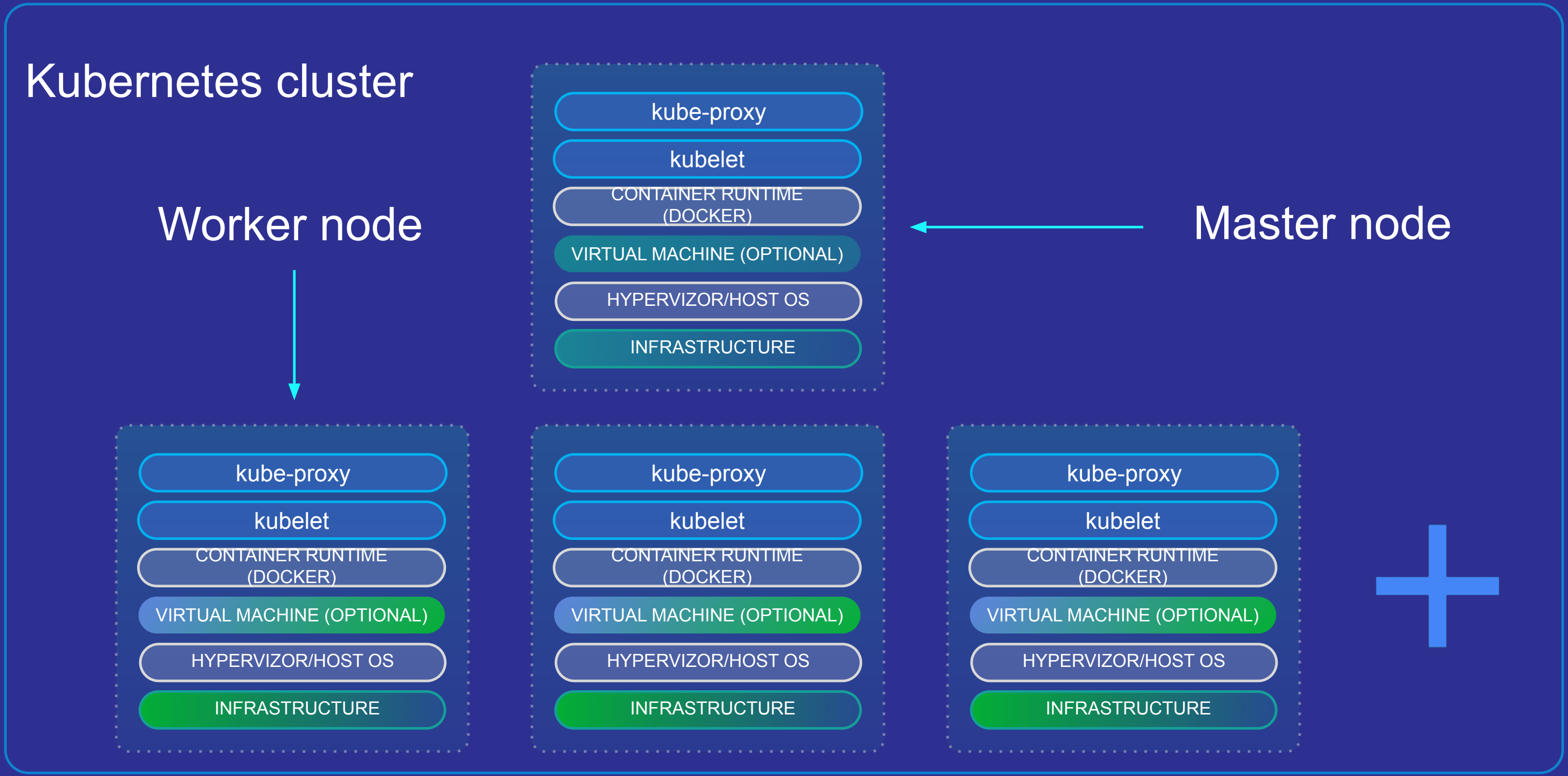
Node



The building blocks – **master node**

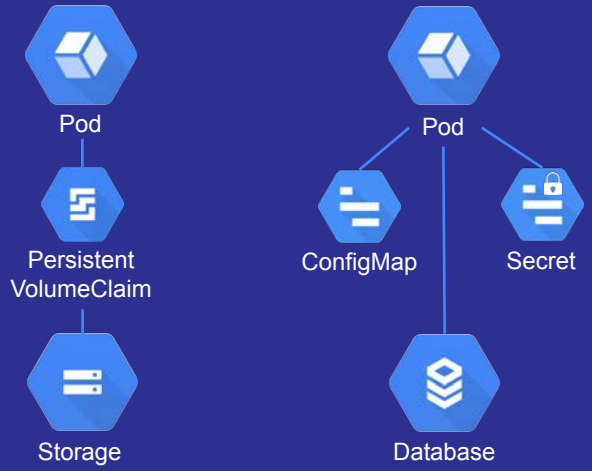


The building blocks – **cluster**



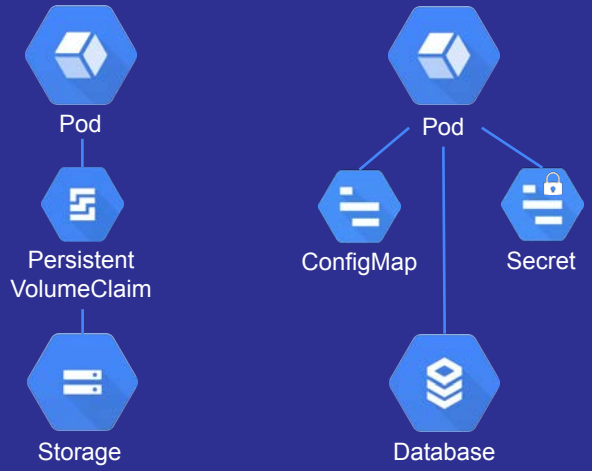
The building blocks – **pods**

- The **pod** is the basic building block of Kubernetes.
- A **pod** contains a group of one or more containers.
- Generally, each **pod** has one container.
- **Pods** handle volumes, secrets and configuration.
- **Pods** are ephemeral – restarted automatically when they die.



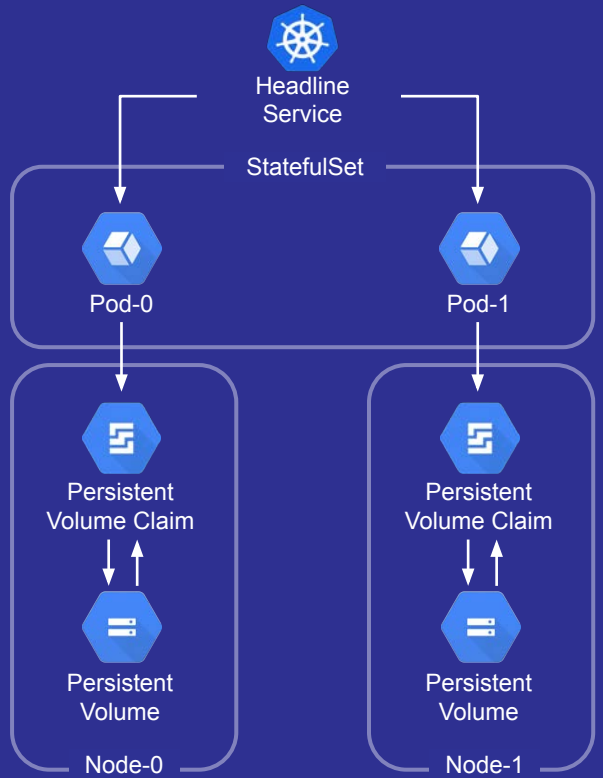
The building blocks – **deployments**

- You can just decide to run pods but when they die, they die.
- A deployment will enable your pod to run continuously.
- Deployments allow you to update a running app without downtime.
- Deployments also specify a strategy to restart pods when they die.



The building blocks – **StatefulSets**

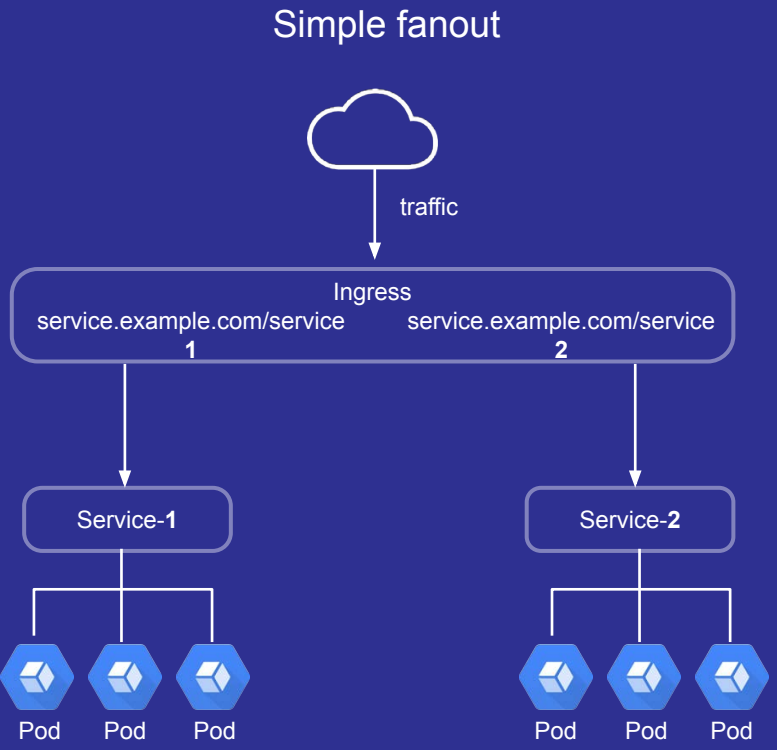
- Does your app require you to keep information about its state?
- A database needs state.
- A StatefulSet's pods are not interchangeable.
- Each pod has a unique, persistent identifier that the controller maintains over any rescheduling.
- Each pod has a unique, persistent identifier that the controller maintains over any rescheduling.



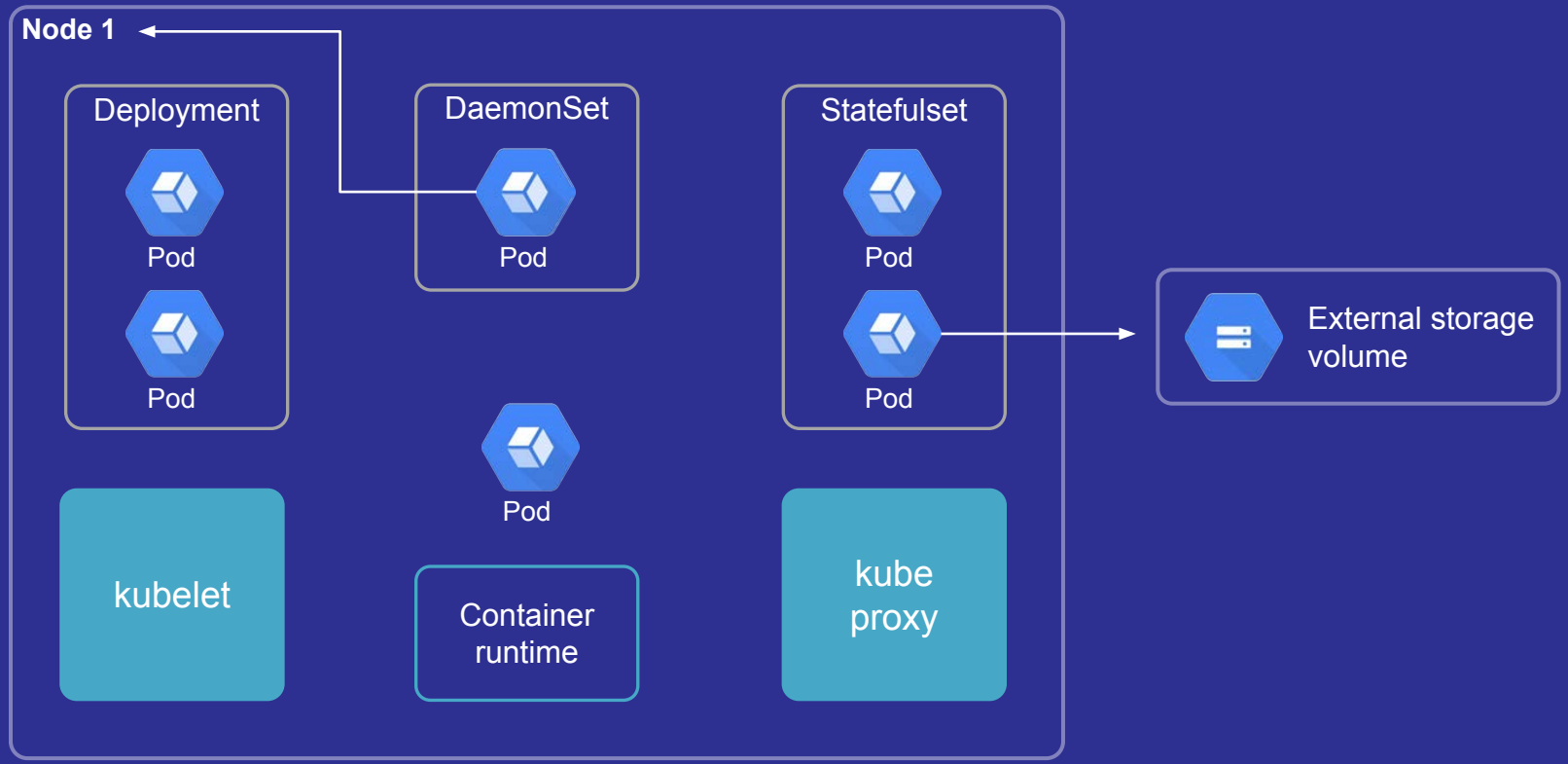
The building blocks – **Kubernetes**

Services:

- A single endpoint to access pods.
- A unified way to route traffic to a cluster and eventually to a list of pods.
- By using a service, pods can be brought up and down without affecting anything.



Workloads



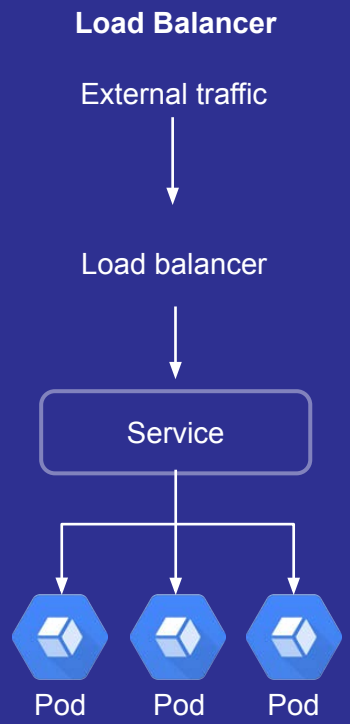
Services

A single endpoint to access pods.

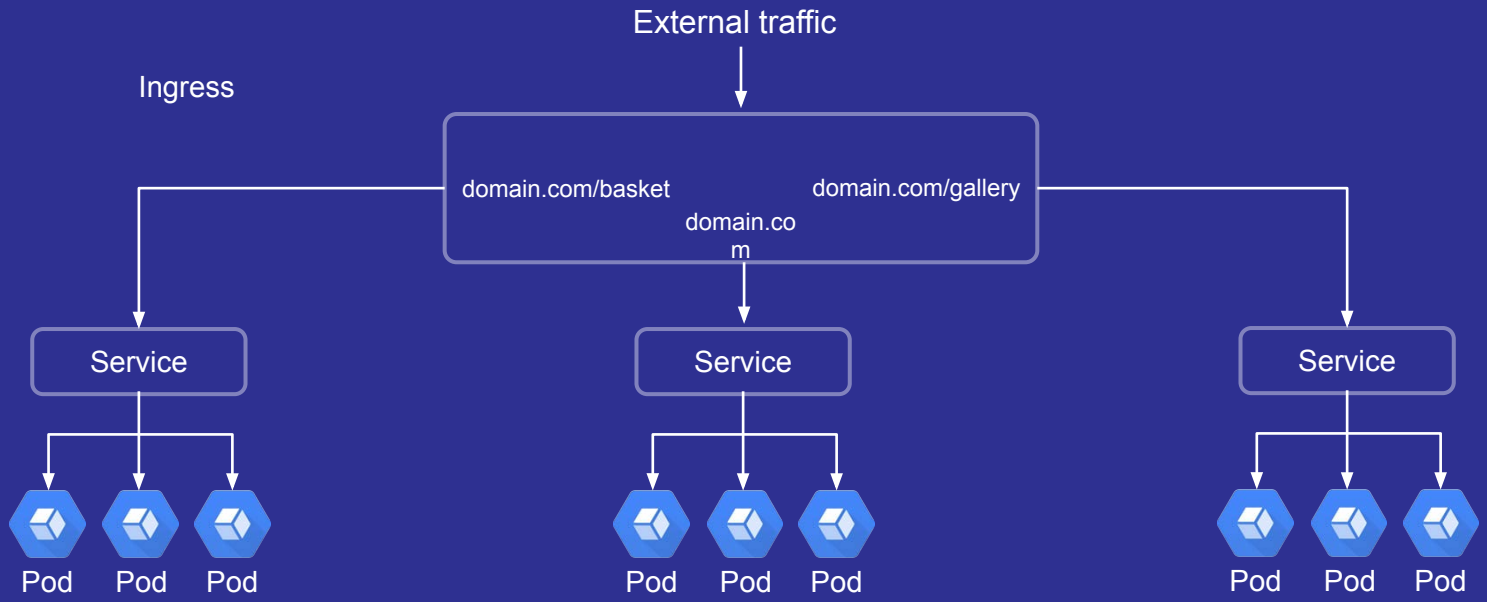
A unified way to route traffic to a cluster and eventually to a list of pods.

By using a service, pods can be brought up and down without affecting anything.

You can also have ClusterIP and NodePort Services.



Services & networking



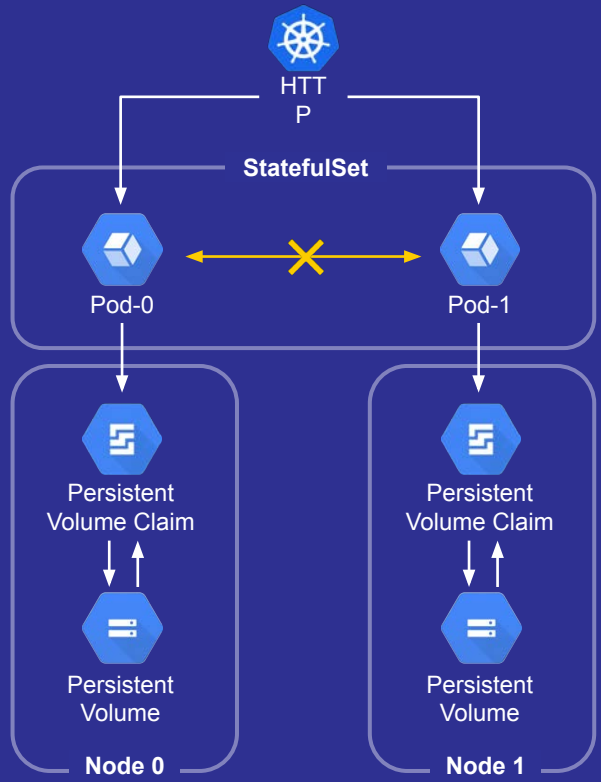
Ingress exposes HTTP and HTTPS routes from outside the cluster to services within the cluster. Traffic routing is controlled by rules defined on the Ingress resource

Networking policies

- Type: network policies are an application-centric construct which allow you to specify how a pod is allowed to communicate with various network "entities" over the network

- By default, if no policies exist in a namespace, then all ingress and egress traffic is allowed
to and from pods in that namespace.

TLDR; control traffic flow at
the IP address or port level (OSI layer 3 or 4).



Conclusion

Kubernetes on vSphere

Kubernetes networking with vSphere

Kubernetes storage with vSphere

vSphere with Tanzu

Day two operations


Thank you!