Moving to Kubernetes

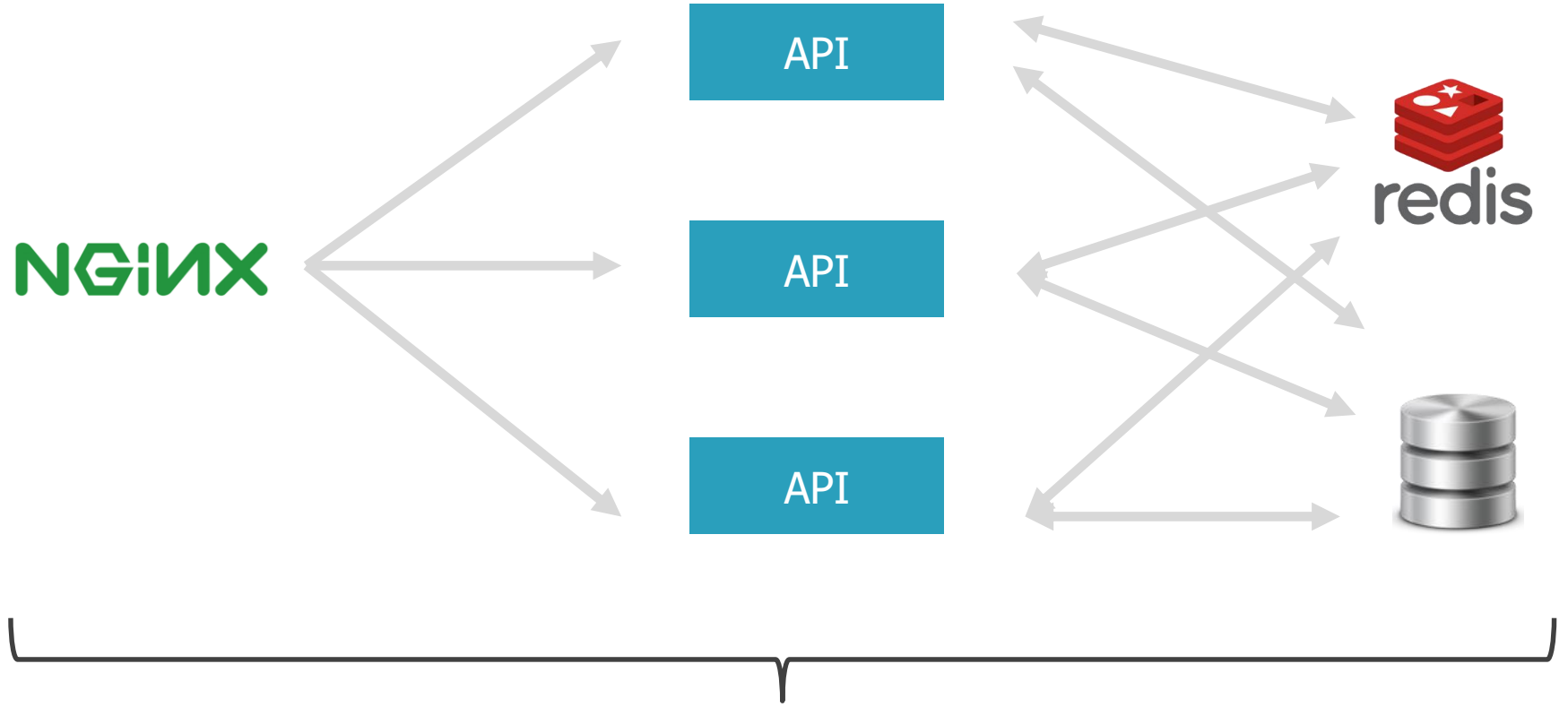
Dan Wahlin
WAHLIN CONSULTING

@danwahlin www.codewithdan.com

Module Agenda

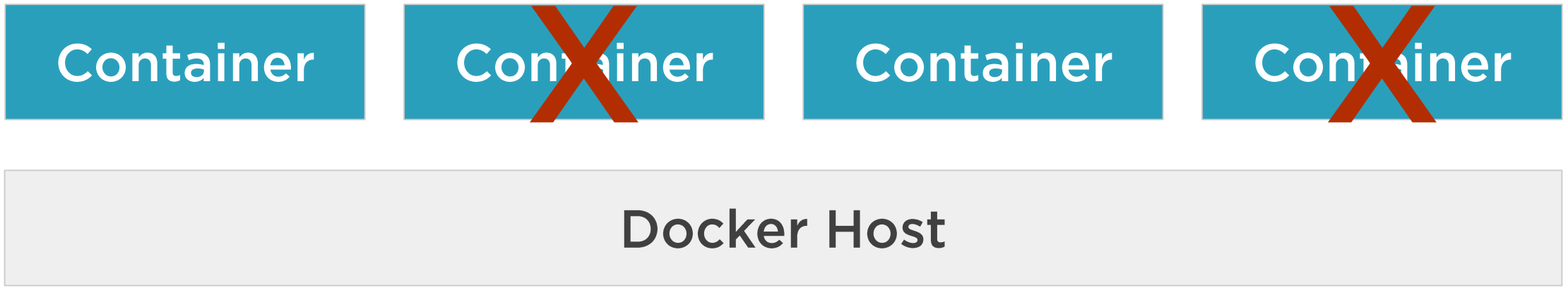
- Beyond Docker Compose
- Introduction to Kubernetes
- Converting from Docker Compose to Kubernetes
- Running Containers in Kubernetes
- Stopping and Removing Containers in Kubernetes

Beyond Docker Compose

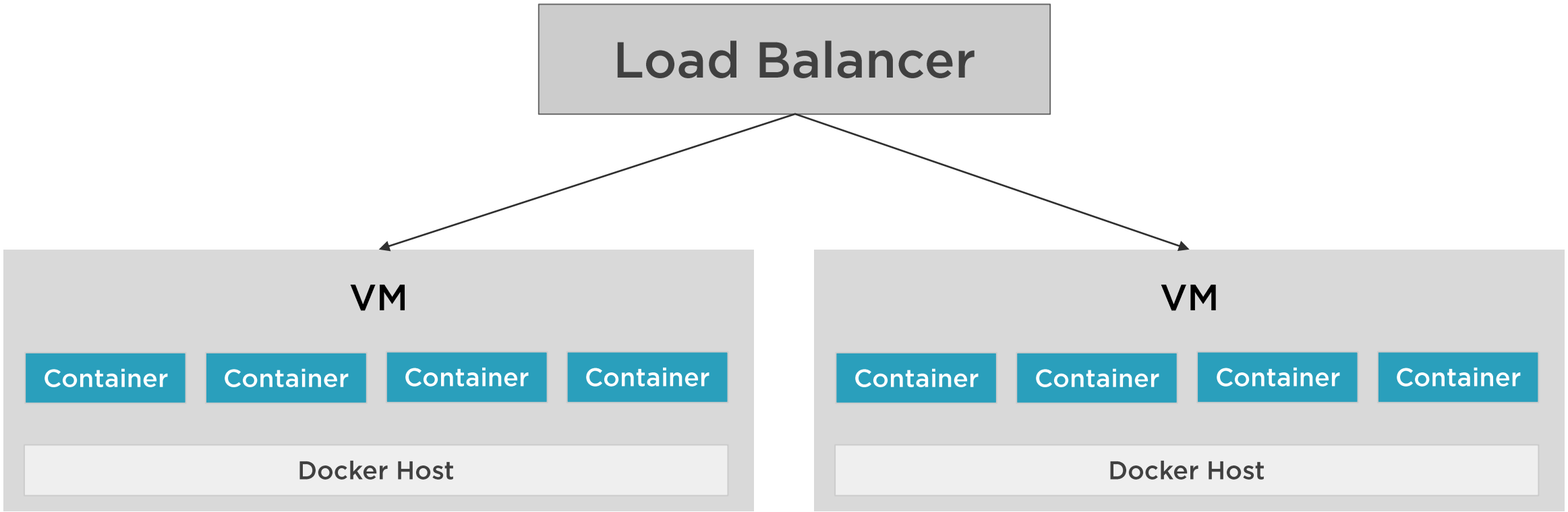


How do you manage all of these containers in test/stage/production?

How Are You Deploying/Managing Containers?



How Do You Scale and Load Balance VMs?



It Would Be Nice if You Could...

- Package up an app, provide a manifest, and let something else manage it for us
- Not worry about the management of containers
- Eliminate single points of failure and self-heal containers
- Have a robust way to scale and load balance containers
- Update containers without bringing down the application
- Have robust networking and persistent storage options

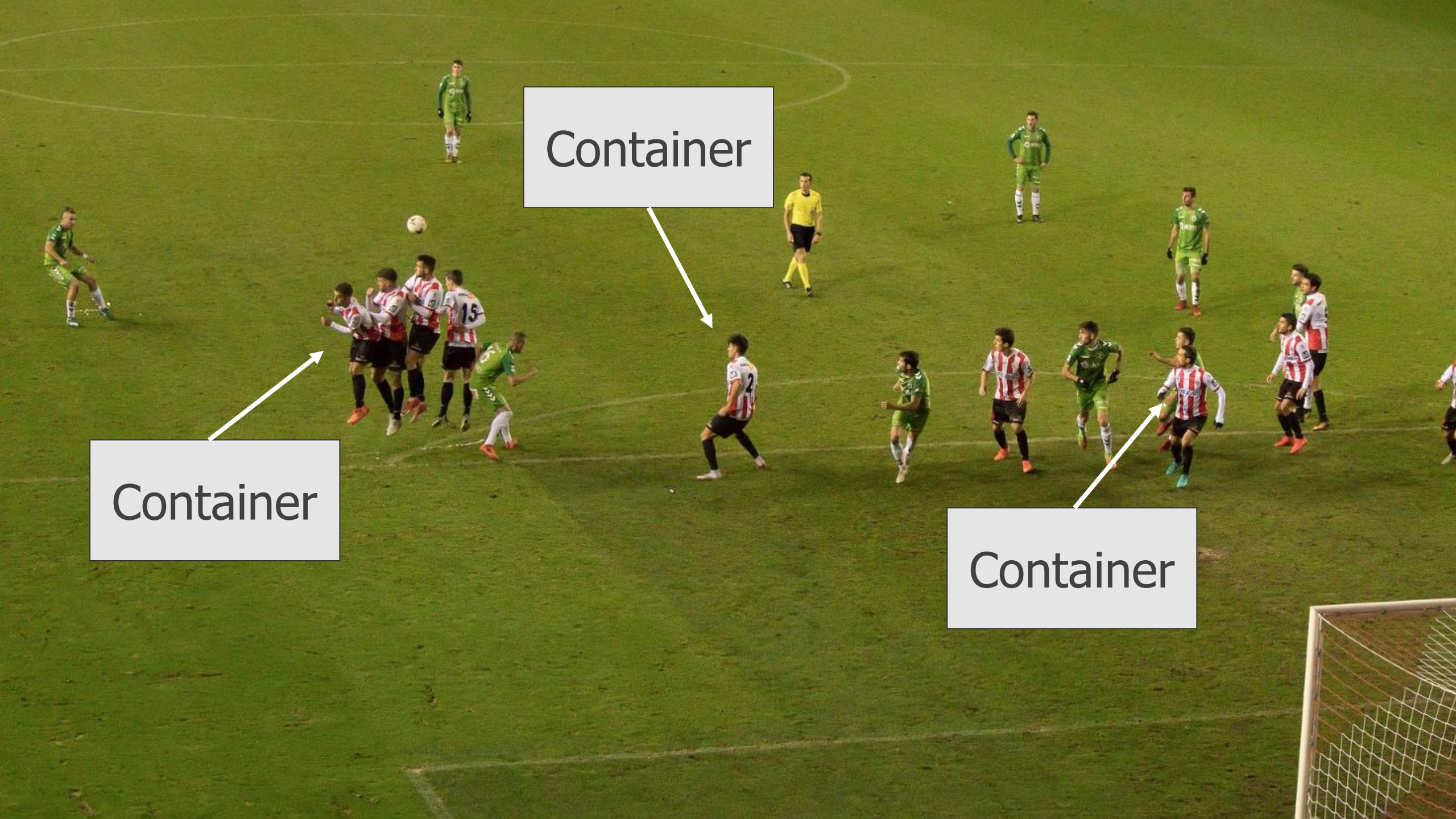
What if we could define the containers we want and then hand it off to a system that manages it all for us?

Welcome to Kubernetes!

Introduction to Kubernetes

"Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications."

https://kubernetes.io


Container
Container
Container

“Kubernetes is the coach of a container team.”


Container
Container
Container
Kubernetes
(conductor)

“Kubernetes is the conductor of a container orchestra.”

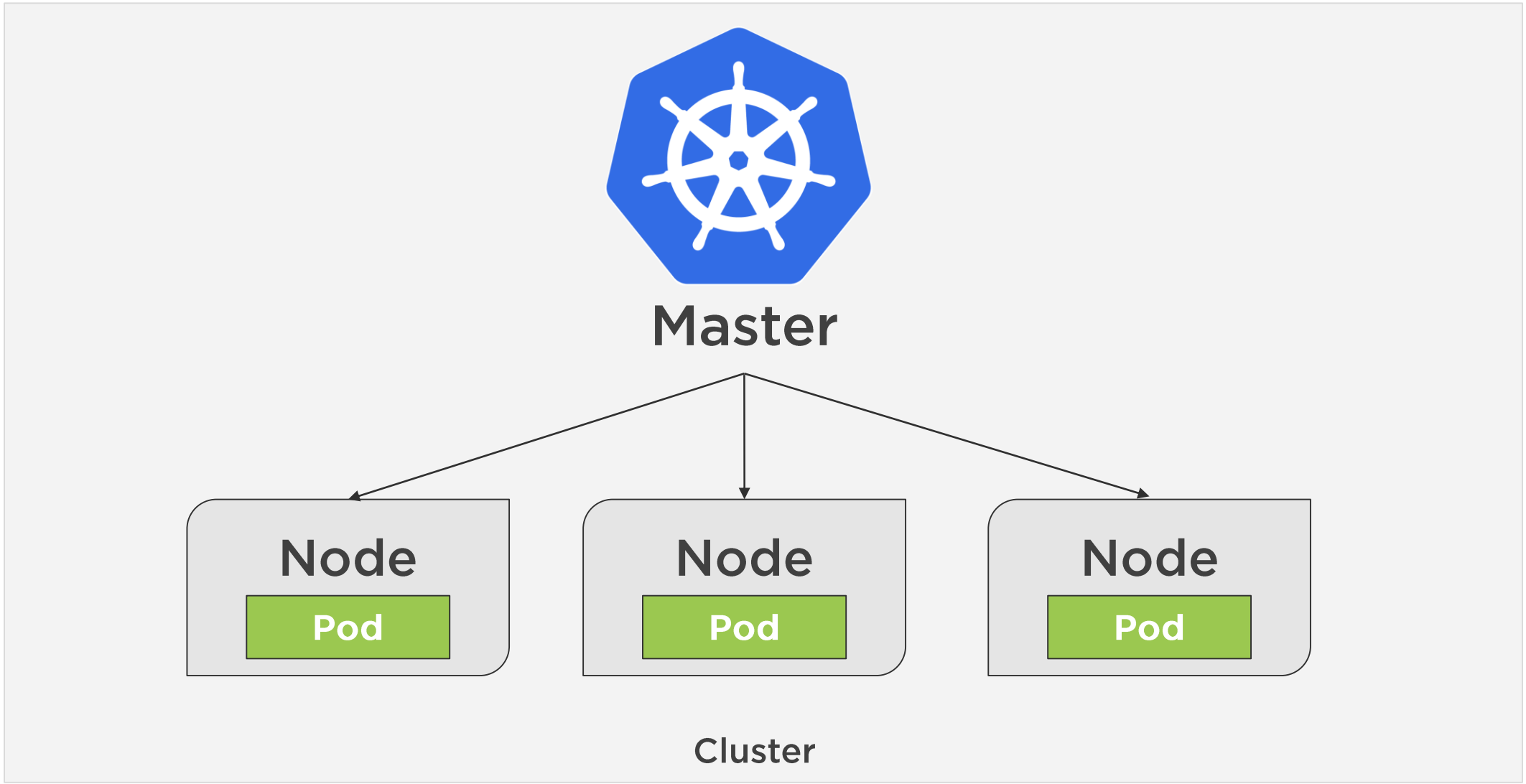
Kubernetes Overview

- Container and cluster management
- Supported by all major cloud platforms
- Provides a declarative way to define a cluster's state using manifest files (YAML)
- Interact with Kubernetes using kubectl

Key Features

- Service Discovery/ Load Balancing
- Storage Orchestration
- Automate Rollouts/Rollbacks
- Manage Workloads
- Self-healing
- Secret and Configuration Management
- Horizontal Scaling
- More

Kubernetes - The Big Picture



Running Kubernetes Locally

Running Kubernetes Locally

Minikube

https://github.com/kubernetes/minikube

Docker Desktop

https://www.docker.com/products/docker-desktop

Key Kubernetes Concepts

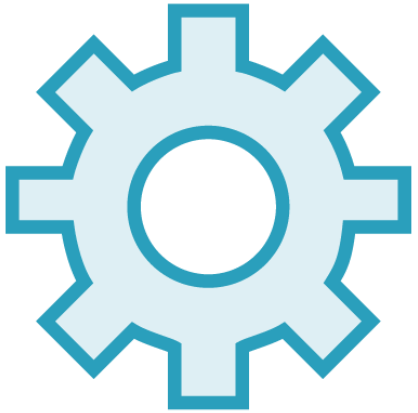
Deployment

Describe desired state

Can be used to replicate pods

Support rolling updates and rollbacks

Service

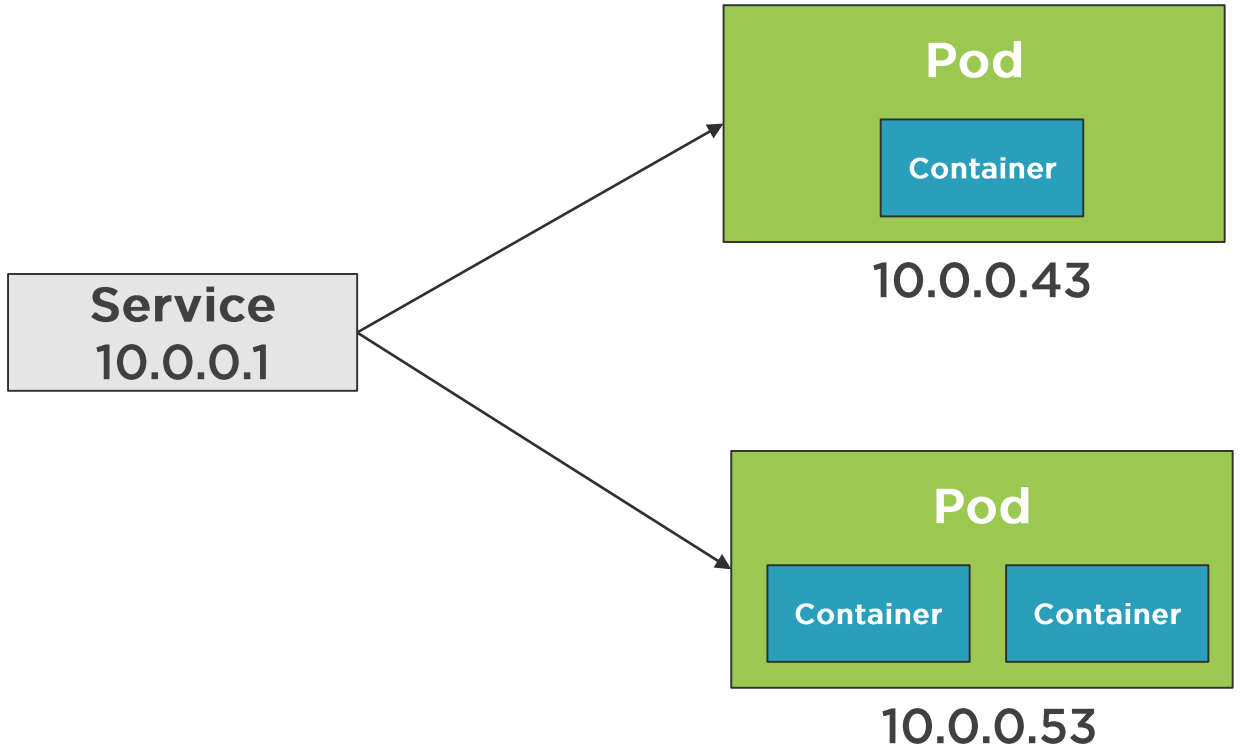


Pods live and die

Services abstract pod IP addresses from consumers

Load balances between pods

Services


Pod
Container
10.0.0.43
Service
10.0.0.1
Pod
Container
Container
10.0.0.53

Converting from Docker Compose to Kubernetes

Migrating from Docker Compose to Kubernetes

Compose on Kubernetes

https://github.com/docker/compose-on-kubernetes

Kompose

http://kompose.io

Running Containers in Kubernetes

kubectl version

kubectl get [deployments | services | pods]

kubectl run nginx-server --image=nginx:alpine

kubectl apply –f [fileName | folderName]

kubectl port-forward [name-of-pod] 8080:80

Stopping and Removing Containers in Kubernetes

kubectl delete –f [fileName | folderName]

Summary

Kubernetes provides a robust solution for automating deployment, scaling, and management of containers

Provides a way to move to a desired state

Relies on YAML (or JSON) files to represent desired state

Nodes and pods play a central role

A container runs in a pod

kubectl can be used to issue commands and interact with the Kubernetes API