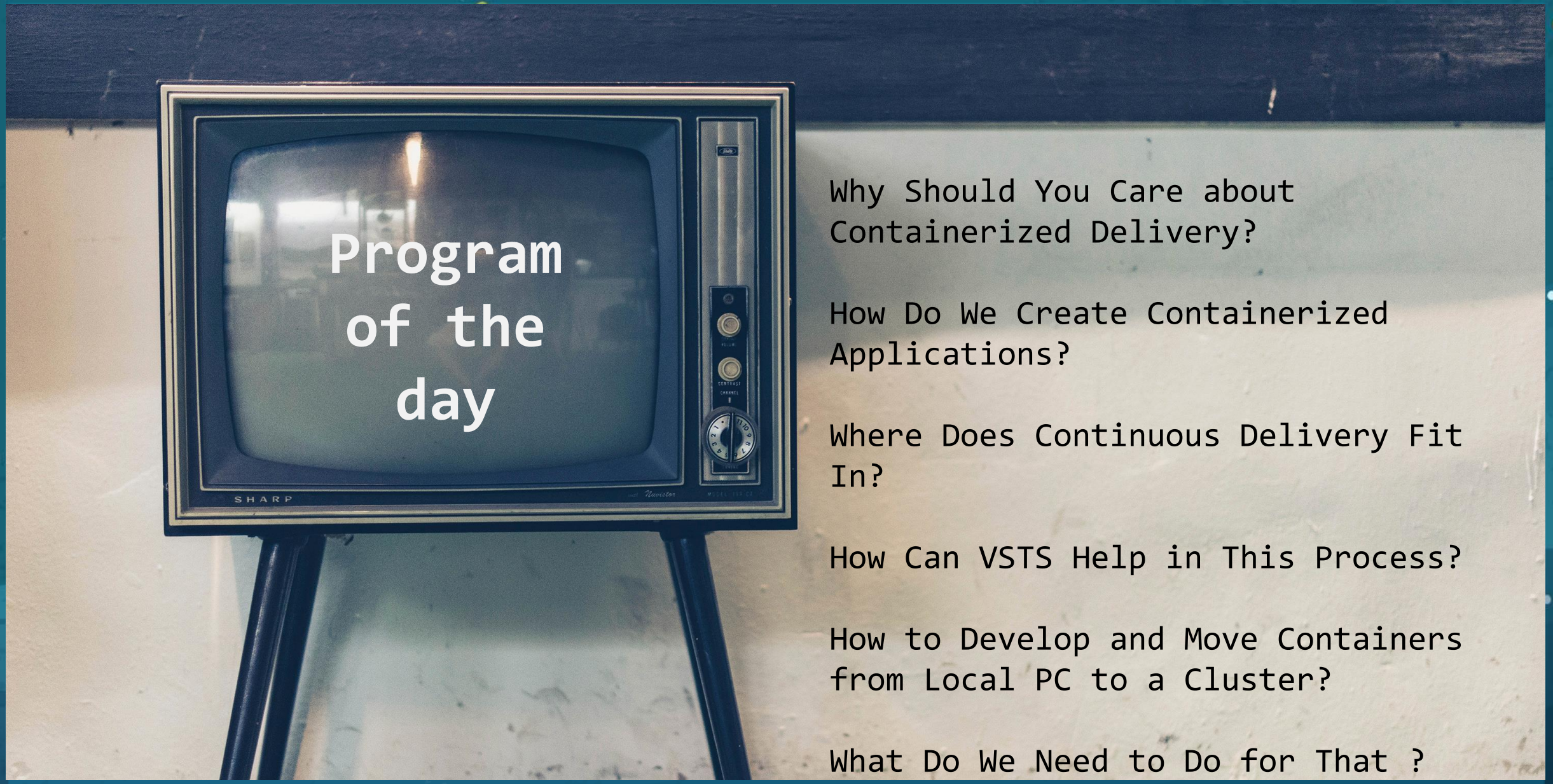
Containerized Delivery with Docker & Visual Studio Team Services

TECHORAMA

René van Osnabrugge
Xpirit

Xpirit

Containers Will Revolutionize the Way We Develop and Deploy Our Applications in the Future



Why Should You Care about Containerized Delivery?

How Do We Create Containerized Applications?

Where Does Continuous Delivery Fit In?

How Can VSTS Help in This Process?

How to Develop and Move Containers from Local PC to a Cluster?

What Do We Need to Do for That ?

Continuous Delivery (CD) is a set of processes, tools and techniques for the rapid, reliable and continuous development and delivery of software.


•Better
•Faster
•Cheaper

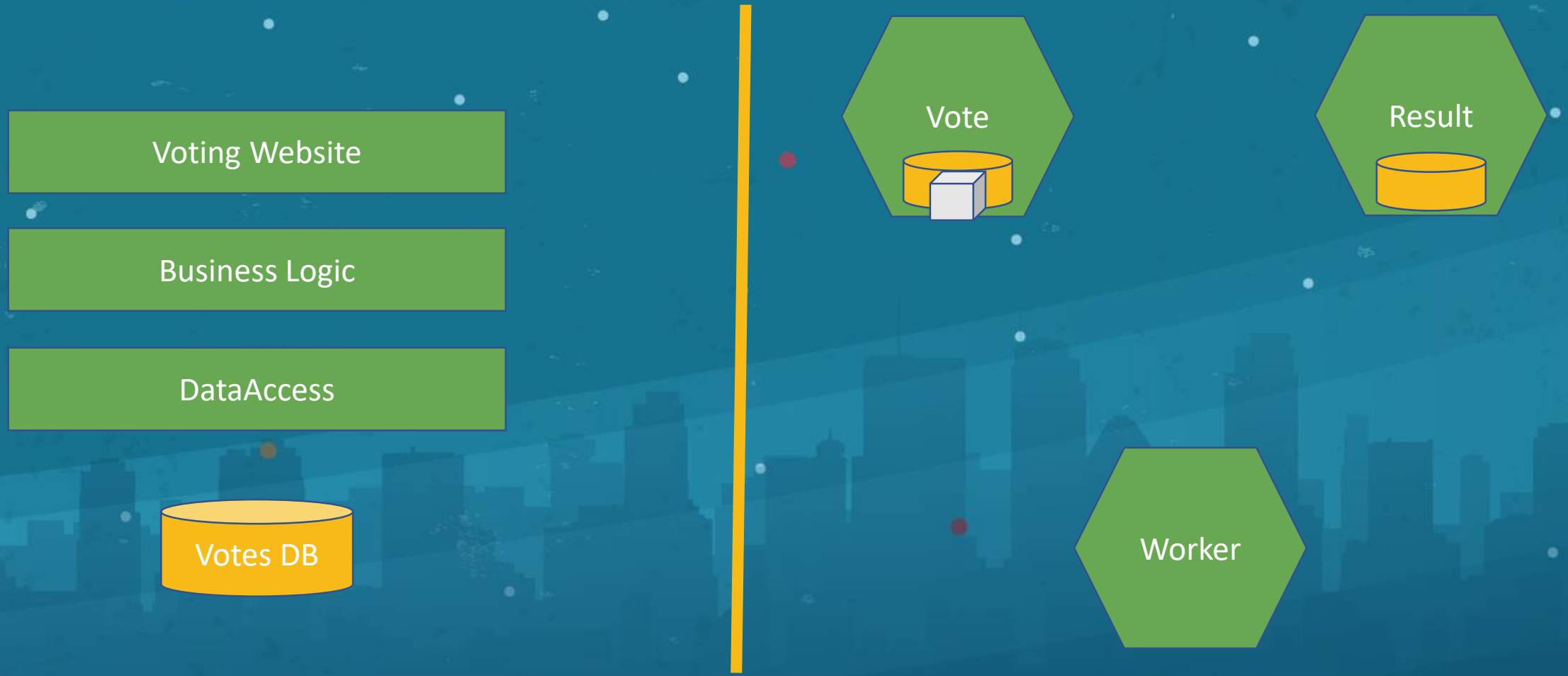
Continuous Delivery Goes Beyond the Pipeline!

TECHORAMA

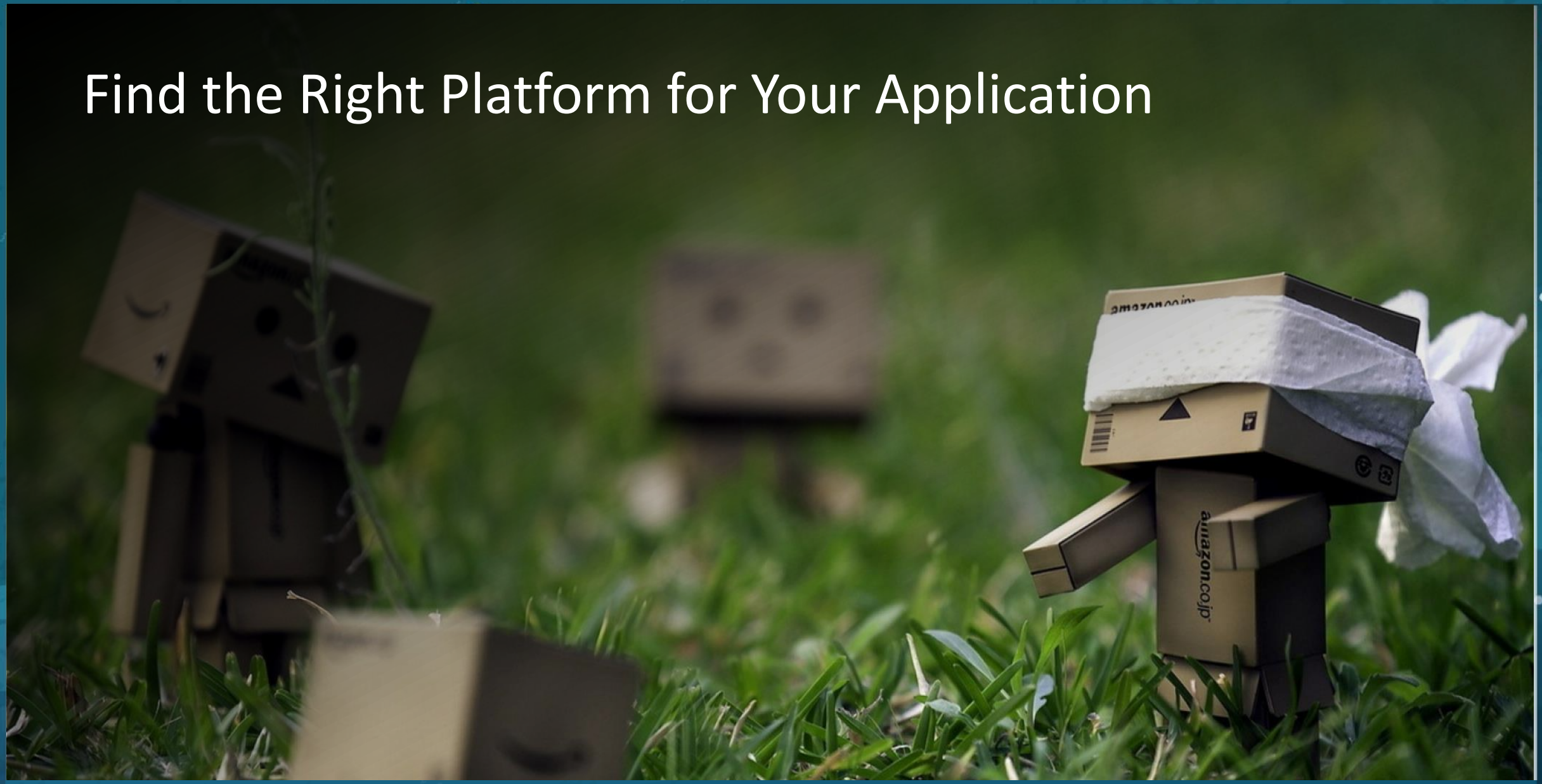
Shipping the Monolith..

TECHORAMA

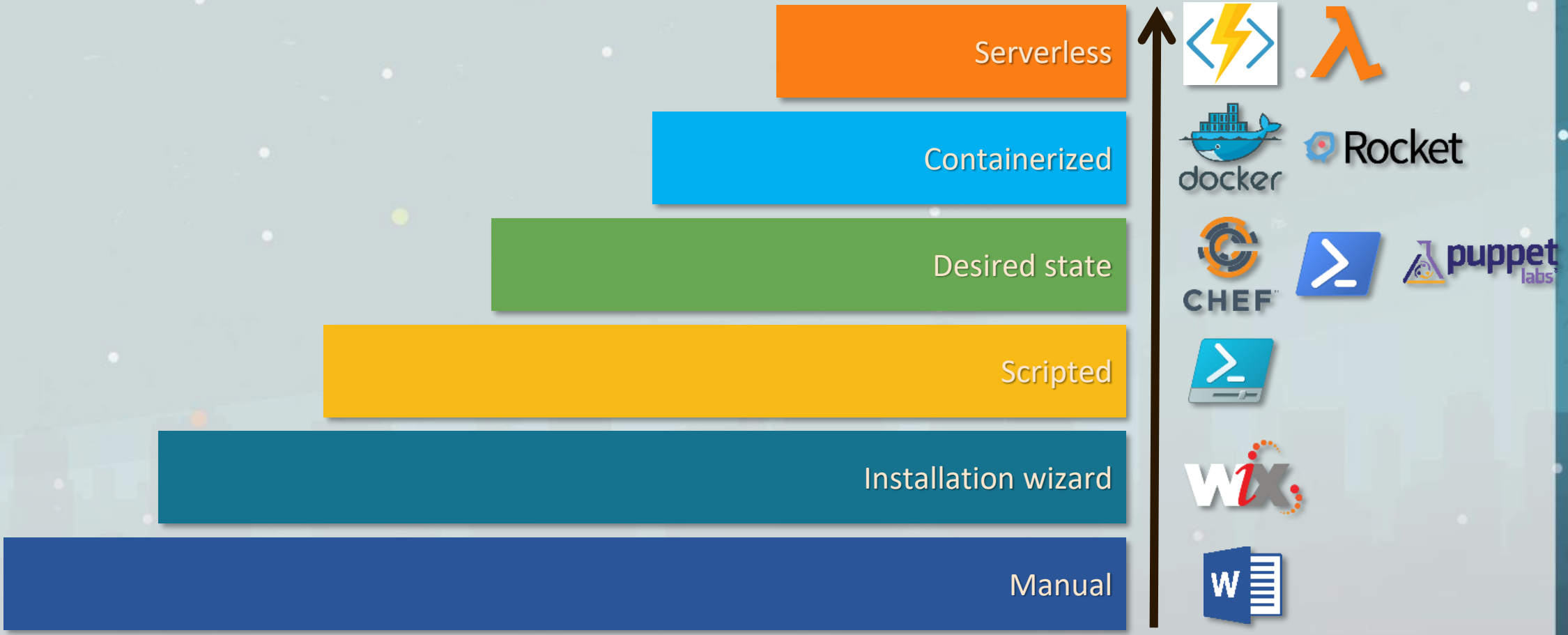
Traditional vs. Microservices Architecture



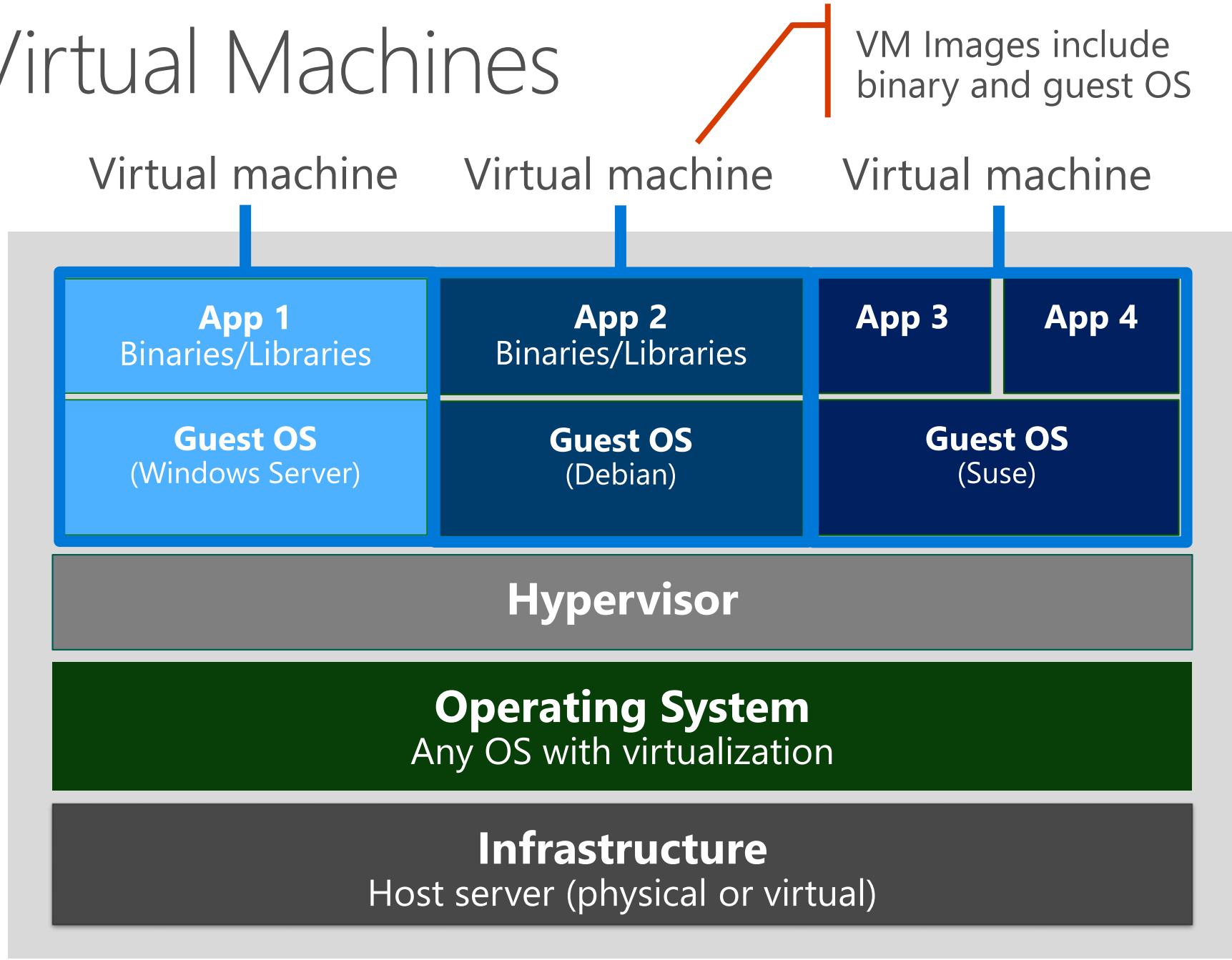
Find the Right Platform for Your Application



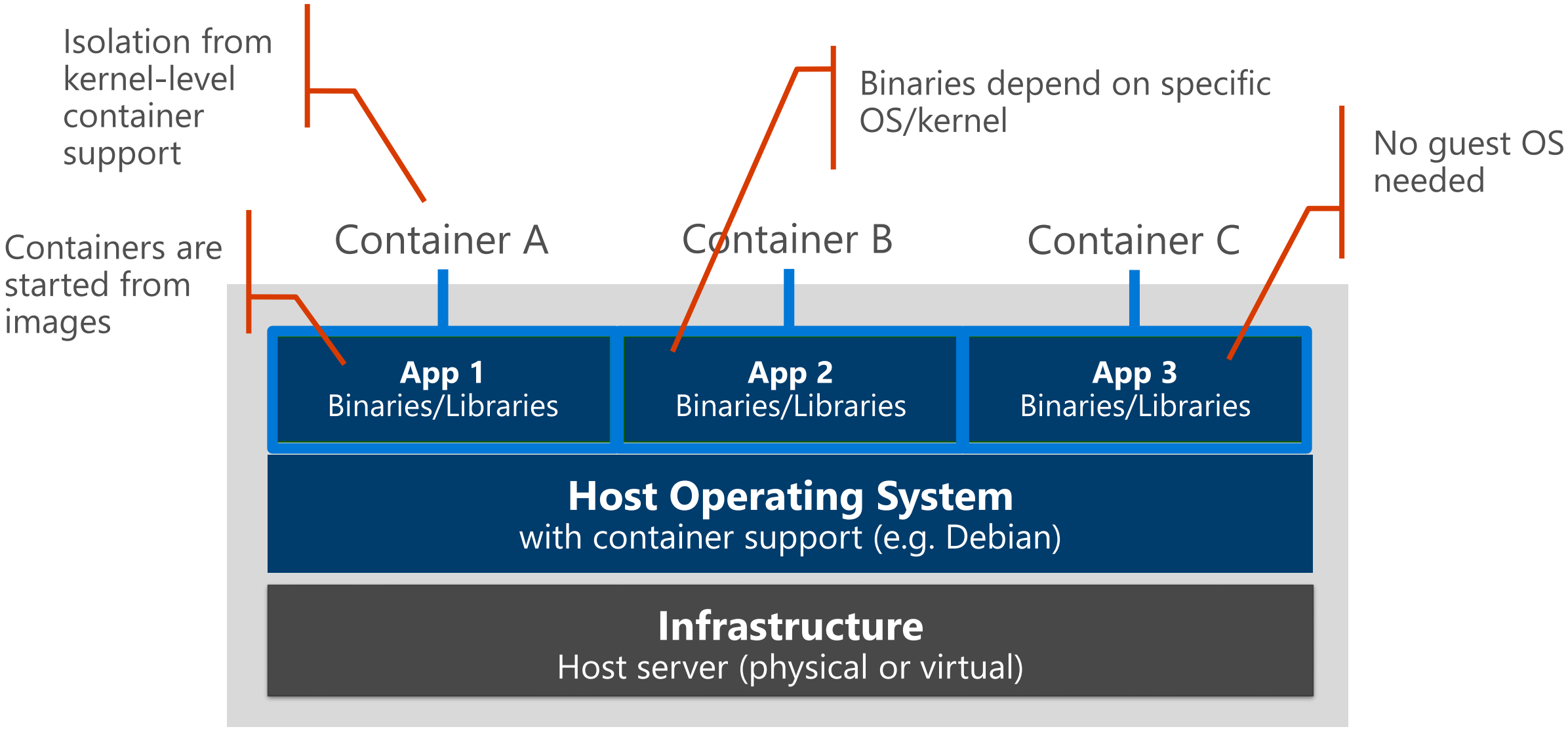
Evolution of Application Delivery



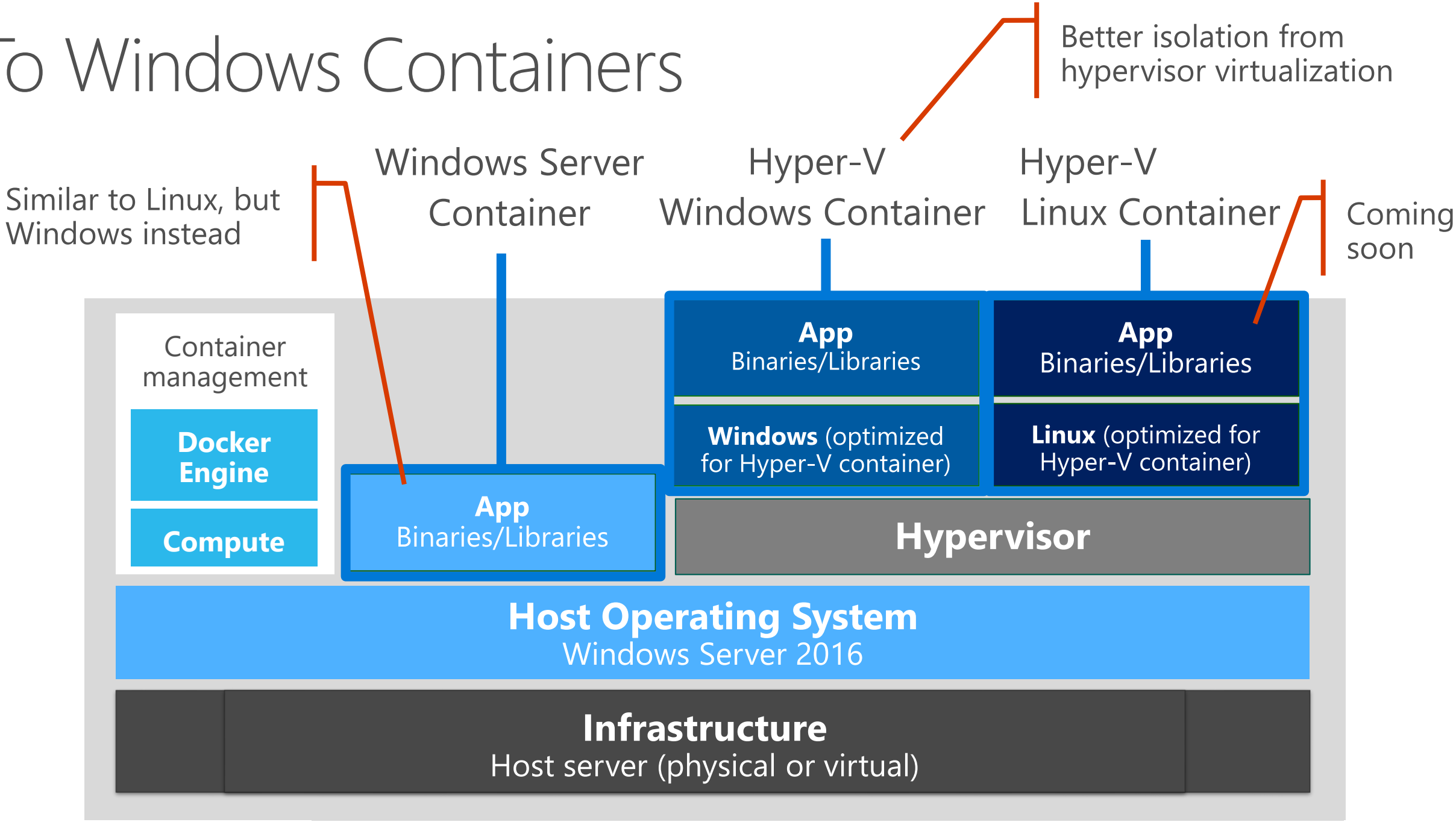
From Virtual Machines



To Linux Containers



To Windows Containers



Docker Image Layers

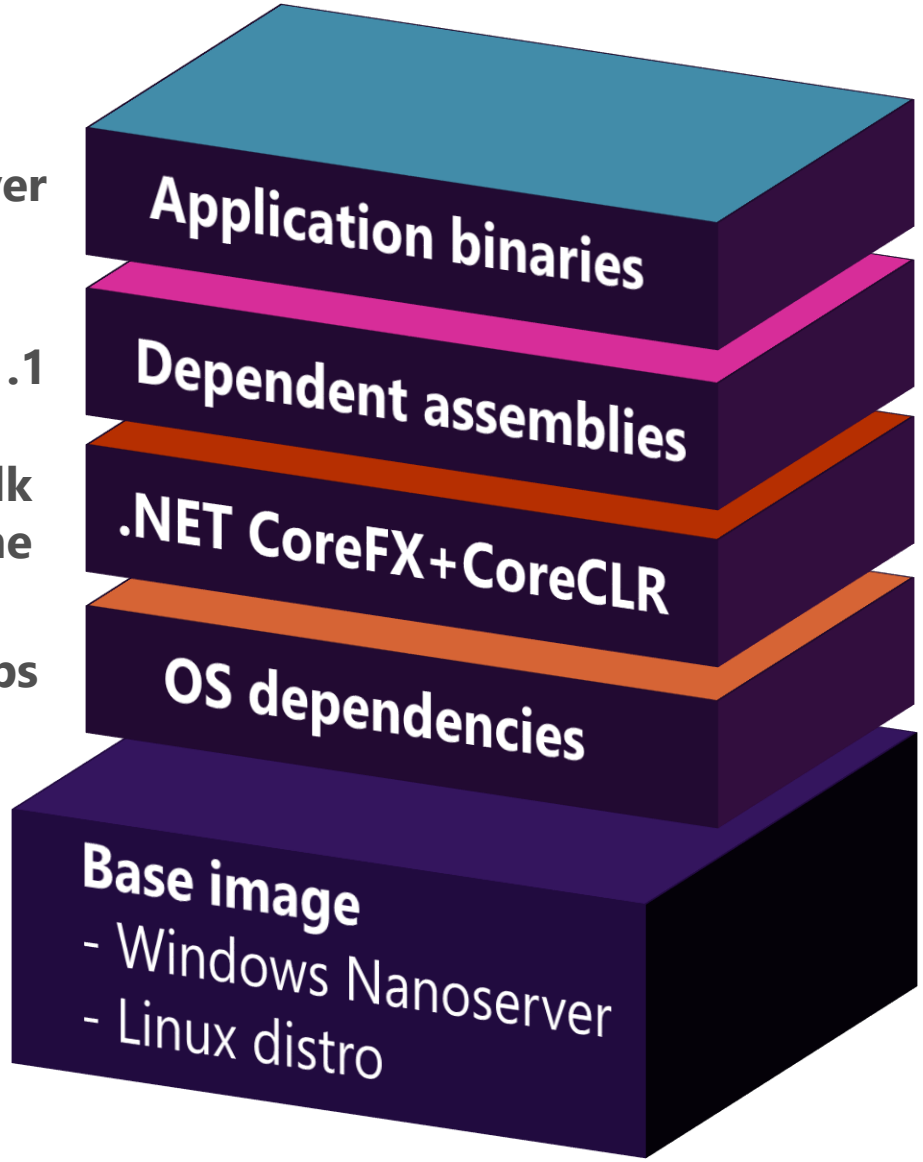
Your application layer

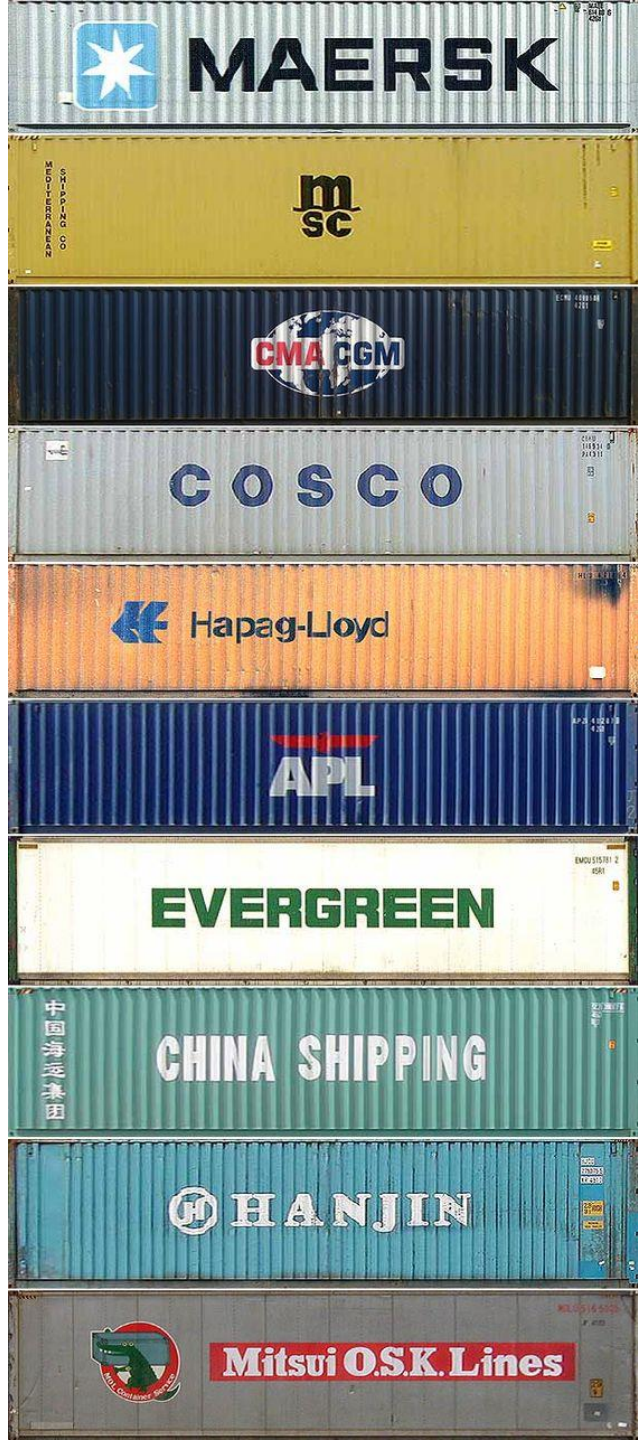
microsoft/aspnetcore:1.1.1

microsoft/dotnet:1.0.0-sdk
microsoft/dotnet:1.1.1-runtime

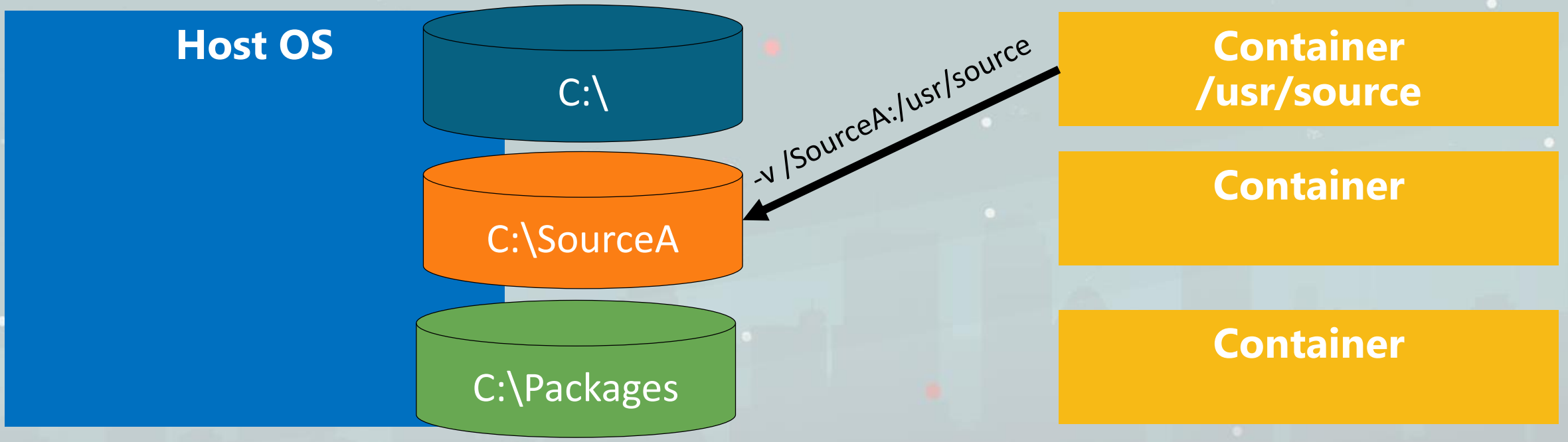
microsoft/dotnet:1.1.1-runtime-deps

microsoft/nanoserver:10.0.14393.1066
e.g. debian/jessie





Volumes Mappings



demo

A short look at Docker and why I like it so much !

The 6-step Guide to Containerized Delivery

1-Move your app into a container

2-Test your composed app locally

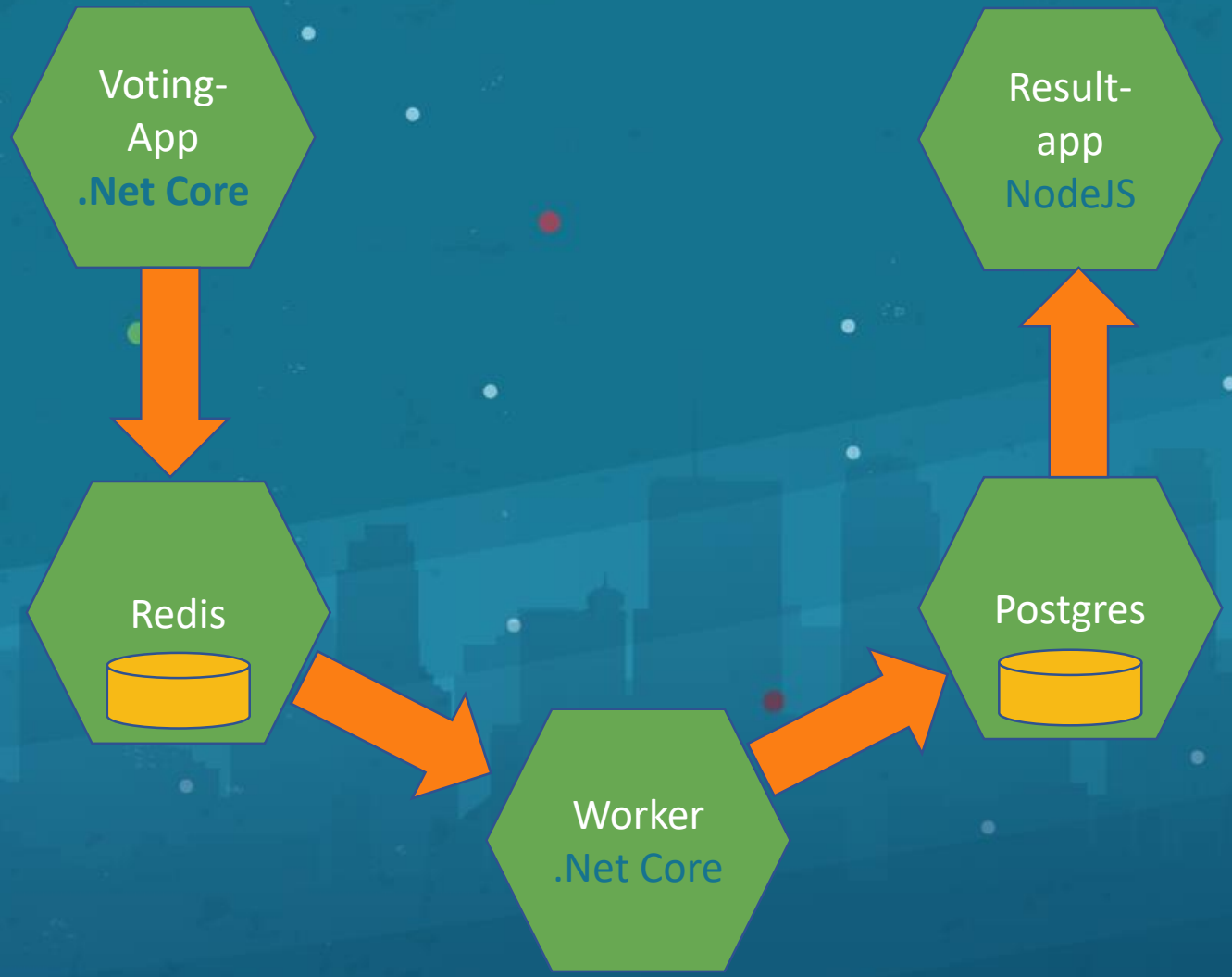
3-Set up a container bakery

4-Publish the container images to a registry

5-Set up a container cluster

6-Release your app on the cluster

Sample App

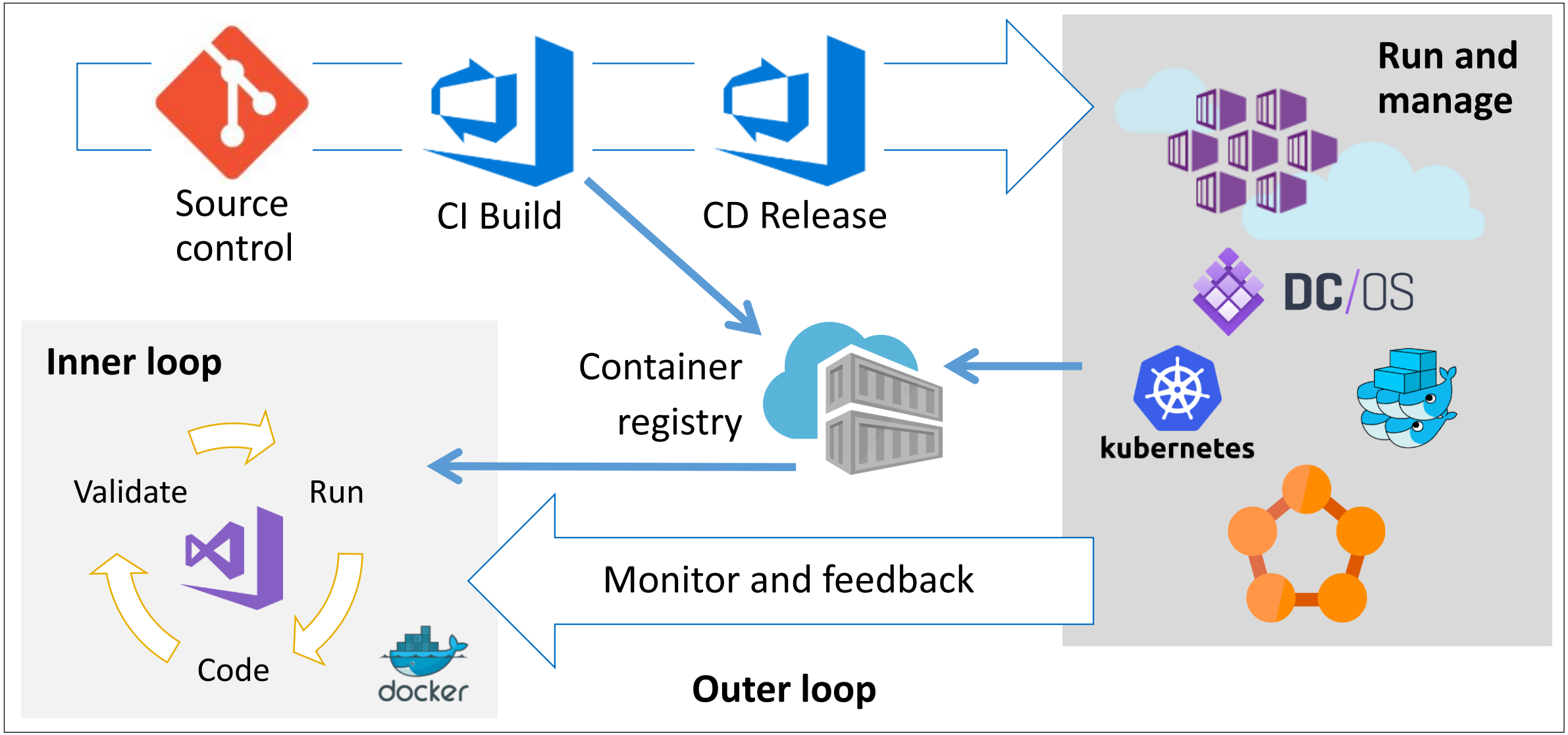

Voting-
App
.Net Core
Result-
app
NodeJS
Redis
Postgres
Worker
.Net Core

demo

1-Move your app to a container

demo

2-Test your composed app locally

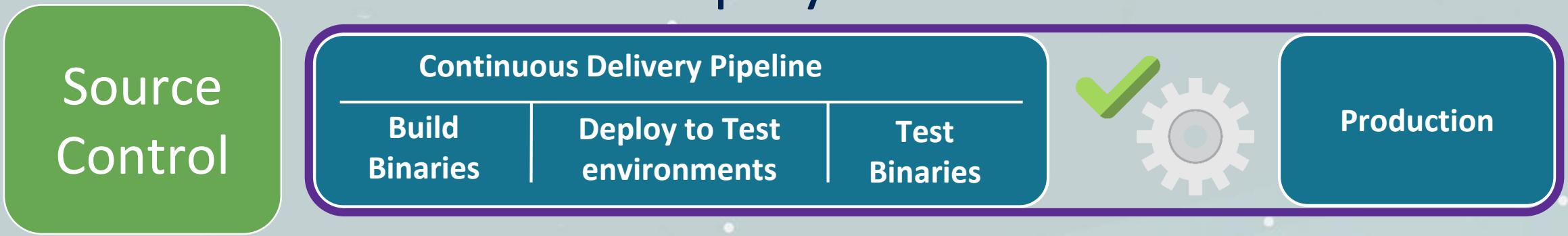

Source control
CI Build
CD Release
Run and manage
DC/OS
kubernetes
Container registry
Inner loop
Validate
Run
Code
docker
Monitor and feedback
Outer loop

The Container registry is key!

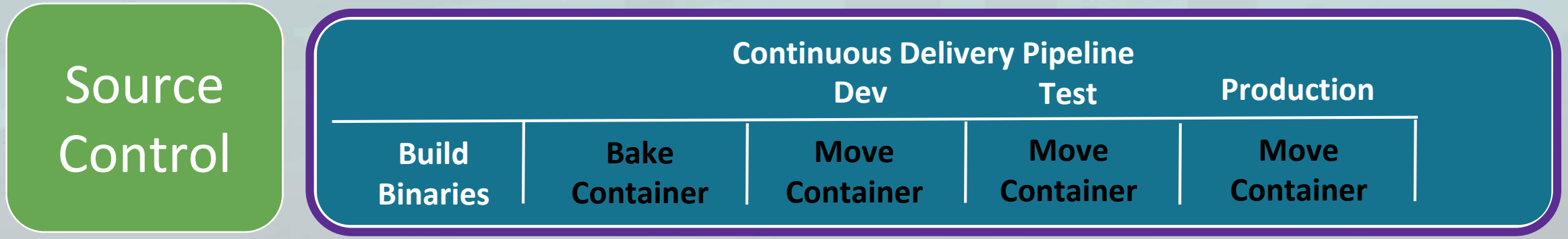
- Stores version of your container
- Functions as distribution point
- Public or private
- Docker Hub/Docker Store
- Azure Container Registry

A New Way of Deployment

Continuous Deployment



Containerized Delivery



demo

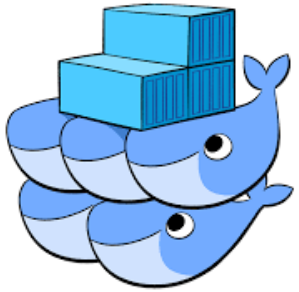
3-Set up a container bakery

4-Publish the container images to a registry

Cluster Orchestrators

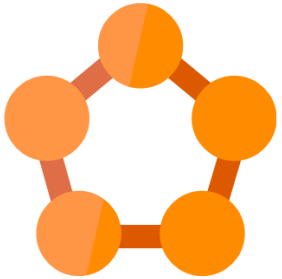
DC/OS

Mesos DC/OS



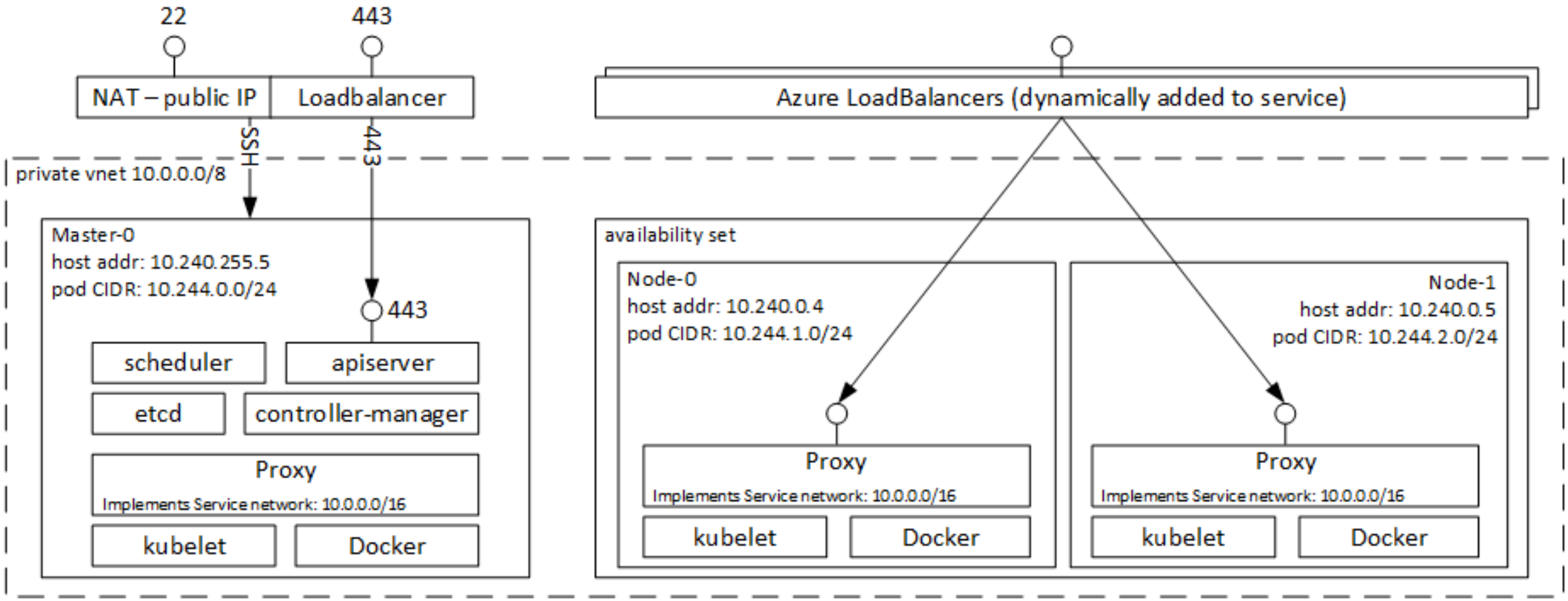
Docker Swarm

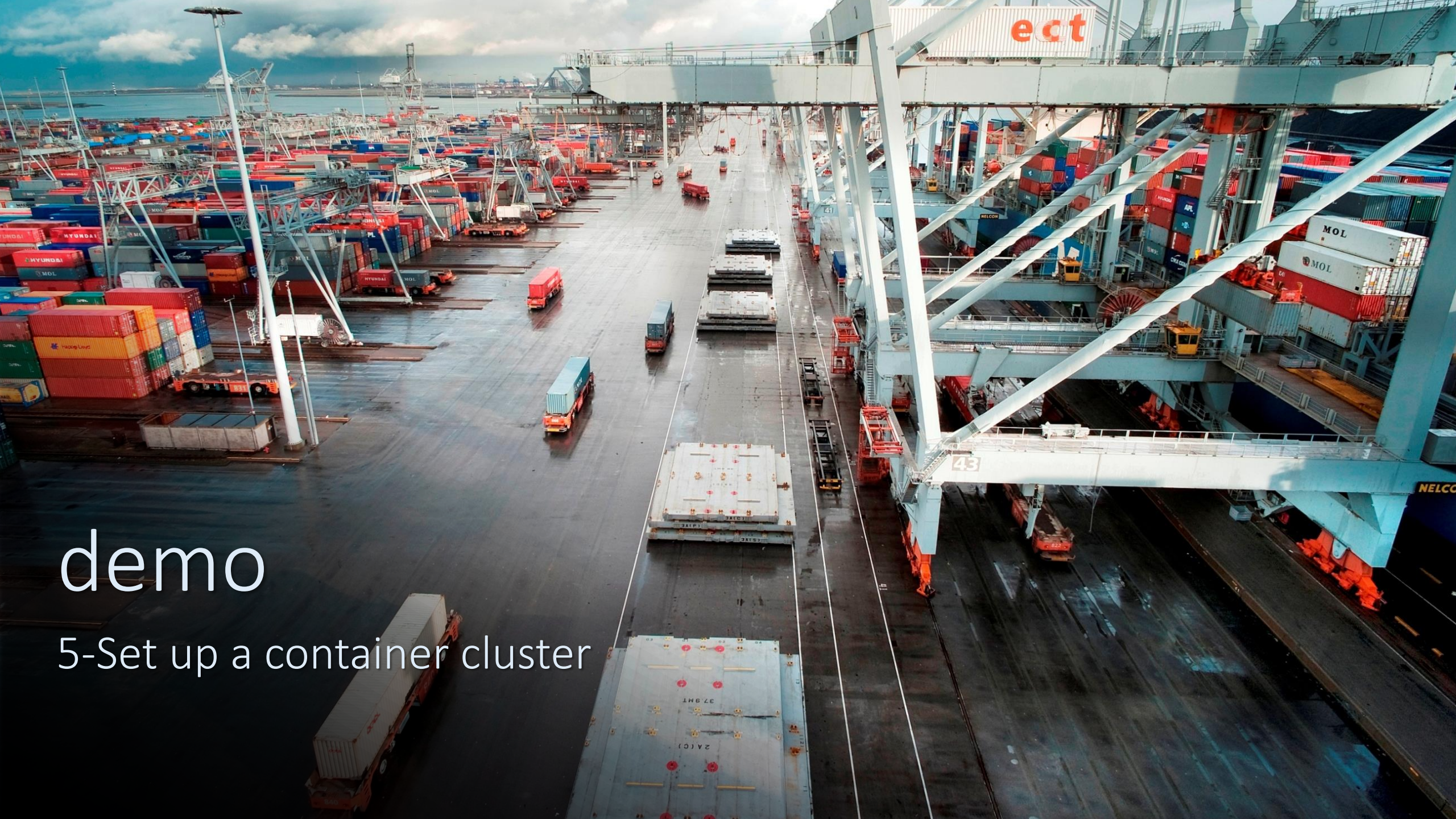
Google Kubernetes



Azure Service Fabric

Clusters of Container Hosts





demo

5-Set up a container cluster

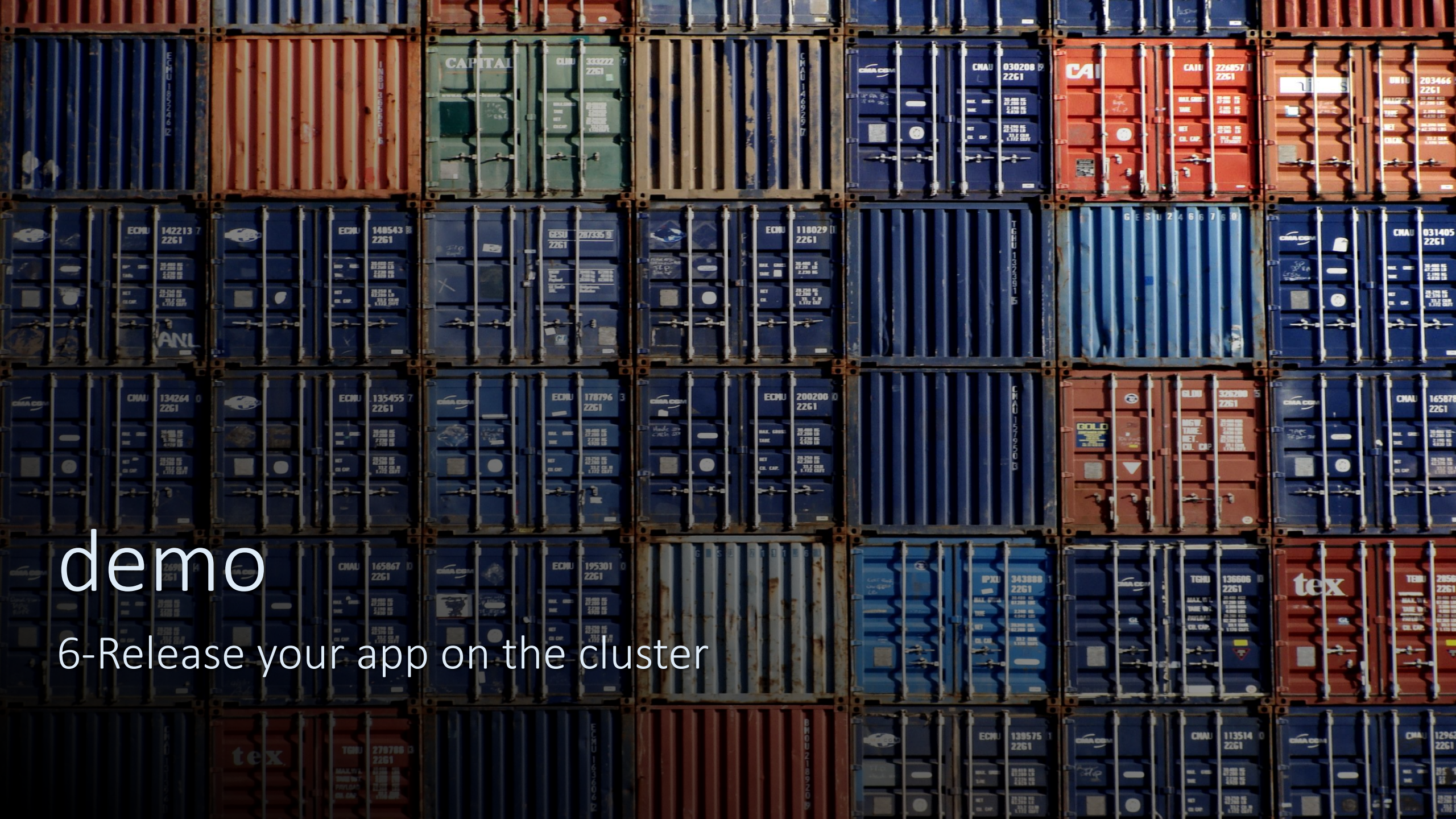
Thin pipelines

Source Control

Bakery

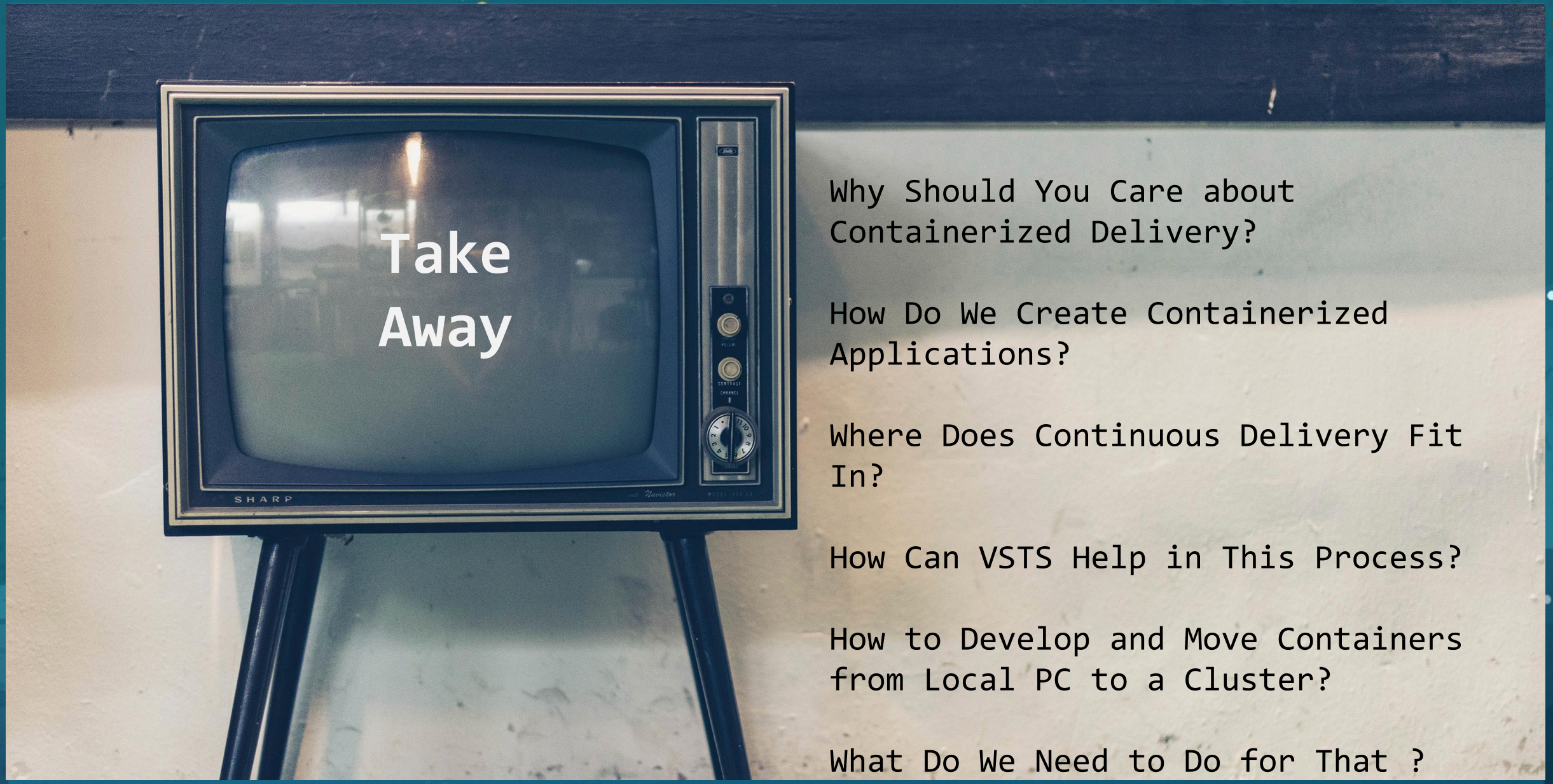
Registry

Initial Deploy



demo

6-Release your app on the cluster



Why Should You Care about Containerized Delivery?

How Do We Create Containerized Applications?

Where Does Continuous Delivery Fit In?

How Can VSTS Help in This Process?

How to Develop and Move Containers from Local PC to a Cluster?

What Do We Need to Do for That ?

René
van Osnabrugge

ALM Consultant at Xpirit

@renevo
rvanosnabrugge@xpirit.com
https://roadtoalm.com
http://xpir.it/rvo-techorama-2017

TECHORAMA

Xpirit

René
van Osnabrugge

ALM Consultant at Xpirit

@renevo
rvanosnabrugge@xpirit.com
https://roadtoalm.com
http://xpir.it/rvo-techorama-2017
http://pages.xpirit.com/magazine-2017

TECHORAMA

Xpirit

do epic Sh!t