Agenda

- What is cloud native?
- Where are we today in the cloud native world?
- The Cloud Native ecosystem.
- Cloud Native adoption.
- Challenges.

What is Cloud Native?

What is Cloud Native?

"A new computing paradigm that is optimized for modern distributed systems environments capable of scaling to tens of thousands of self healing multi-tenant nodes"

-Cloud Native Computing Foundation

Pillars of Cloud Native

Pillars of Cloud Native: Devops

"DevOps is the practice of operations and development engineers participating together in the entire service lifecycle, from design through the development process to production support."

-The Agile Admin blog
https://theagileadmin.com/what-is-devops/

Pillars of Cloud Native: Devops

- Generally based on principles of CALMS
- Based on the ideas of Automation, Measurement, Sharing
- Emphasis on a Collaborative culture in organizations
- Shifting operations more to the left.
  - Operations teams does more than just “server management”
  - Uses same techniques as developers for systems work

Pillars of Cloud Native: CD

- Origins in the automation segment of Devops
- Similar to Continuous Integration
  - Frequent code commits into source control
  - Run automated builds against each code commit
  - Result= Detecting errors quicker
- Continuous Delivery
  - Step 1: Continuous Integration
  - Release code builds to end users
  - Result= ship software quicker to end users

Pillars of Cloud Native: Microservices

Microservices Design

- Start with Twelve-Factor App design
- https://12factor.net
- Based on the principals of software design and deployment at Heroku
- Development best practice that synergizes with devops engineers

Twelve Factor App

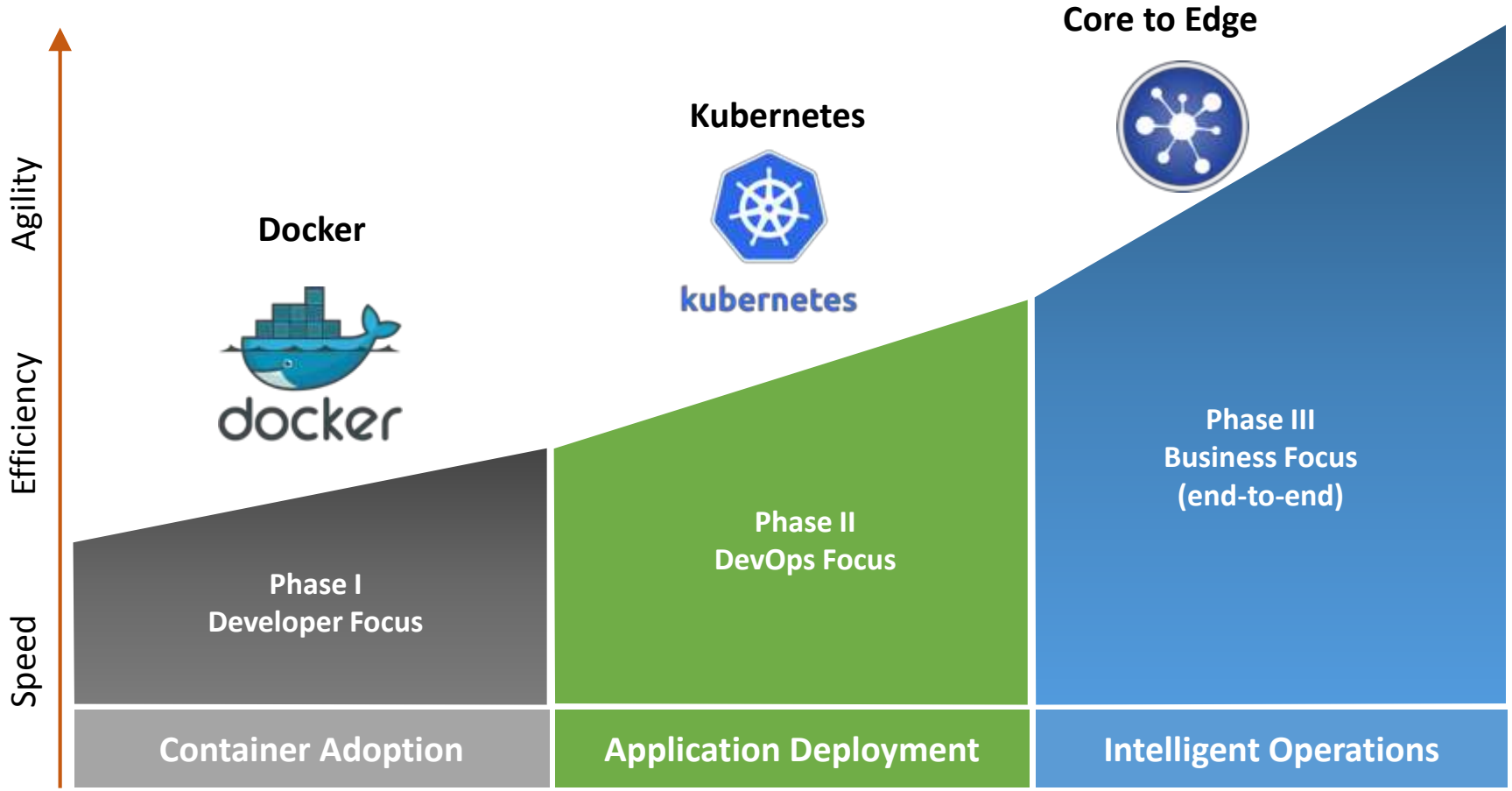
- Principles:
  - Declarative formats for Automation
  - Clean Contracts with underlying systems
  - Able to deploy to modern cloud platforms
  - Keep your prod and dev systems similar
  - Easily scale up without changes to architecture, tooling, development etc.

- 12 factor apps enable easy container integration

Pillars of Cloud Native: Containers

- Way to package applications
- Fits really well as a packaging strategy for microservices
- Not a new concept
- Popularized by the growth of Docker, and Kubernetes

Pillars of Cloud Native: Containers



Focus
Applications
Automation
Community

Developer adoption
Dev/Test apps
Simple orchestration
Individual developers

DevOps deployment
Production apps
Advanced orchestration
Teams & lines of business

End-to-end integration
Digital business apps
Serverless, DevSecOps, & ML
Cloud native enterprises

1	React.js	468 mentions
2	Kubernetes	335
3	Docker	252
4	Linux	240
5	Tensorflow	226
6	Vue.js	192
7	Laravel	170
8	Angular.js	123
9	Python	95
10	Ruby on Rails / Android	84 (tied)

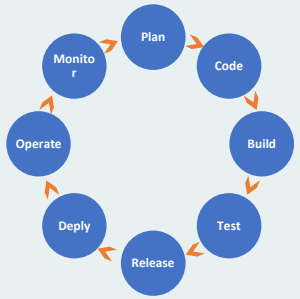
Developer Trends in the Cloud: Open source
Digital Ocean Survey, October 2018

Respondents=4300

https://www.digitalocean.com/currents/october-2018/

Evolution of Development and Deployment

	Development Process	Application Architecture	Deployment and Packaging	Application Infrastructure
~ 1980 ~ 1990	Waterfall	Monolithic	Physical Server	Datacenter
~ 2000	Agile	N-Tier	Virtual Servers	Hosted
~ 2010 Now	DevOps	Microservices	Containers	Cloud



Cloud Native Usecases

Key Container Use Cases

	Share	Container Use Cases	Orchestration Use Cases
Development	65%	Developer productivity; Consistent appstacks in Dev, Test & Production	Automated deploys to accelerate application release cadence
CI/CD/DevOps	48%	Containerized dependencies; Container registries;	Rolling updates and reversals
Operations	41%	Standardized environments for dev, testing and operations	Resilient, self-healing systems; High Availability; Elastic Scalability
Refactor Legacy Apps	34%	Refactor from N-tier to portable containerized applications	Run distributed, stateful apps on scale-out infrastructure
Migrate to Cloud	33%	Move entire appstacks and see them run identically in the cloud	Cloud bursting; Reduce infrastructure costs by avoiding over-provisioning
New Microservice Apps	32%	Create small purpose-built services that can be assembled to scalable custom applications	Dynamically manage large-scale microservices infrastructure

SOURCE: THE EVOLUTION OF THE MODERN SOFTWARE SUPPLY CHAIN, DOCKER SURVEY 2016


New Technology??
Must Adopt!!

Business and Engineering

The Business Case for Cloud Native

- Quicker Time to Deliver
- Modernizing present day applications
- Develop new applications quickly
- Improve speed of innovation

Quicker Time To Deliver

- Containers + Microservices allows for a common language between your development and operations teams
  - Shared Understanding…
  - Allows for IT in general to practice a devops culture
  - Less friction between various teams in the organization
- Practicing Continuous Delivery allows you to ship faster
  - Process of making changes becomes easily
  - Reduces perceived risk of making changes

Modernizing present day applications

- Shipping applications in containers reduces dependencies on underlying infrastructure
- As a result, previous on premise applications can be exported to the cloud.

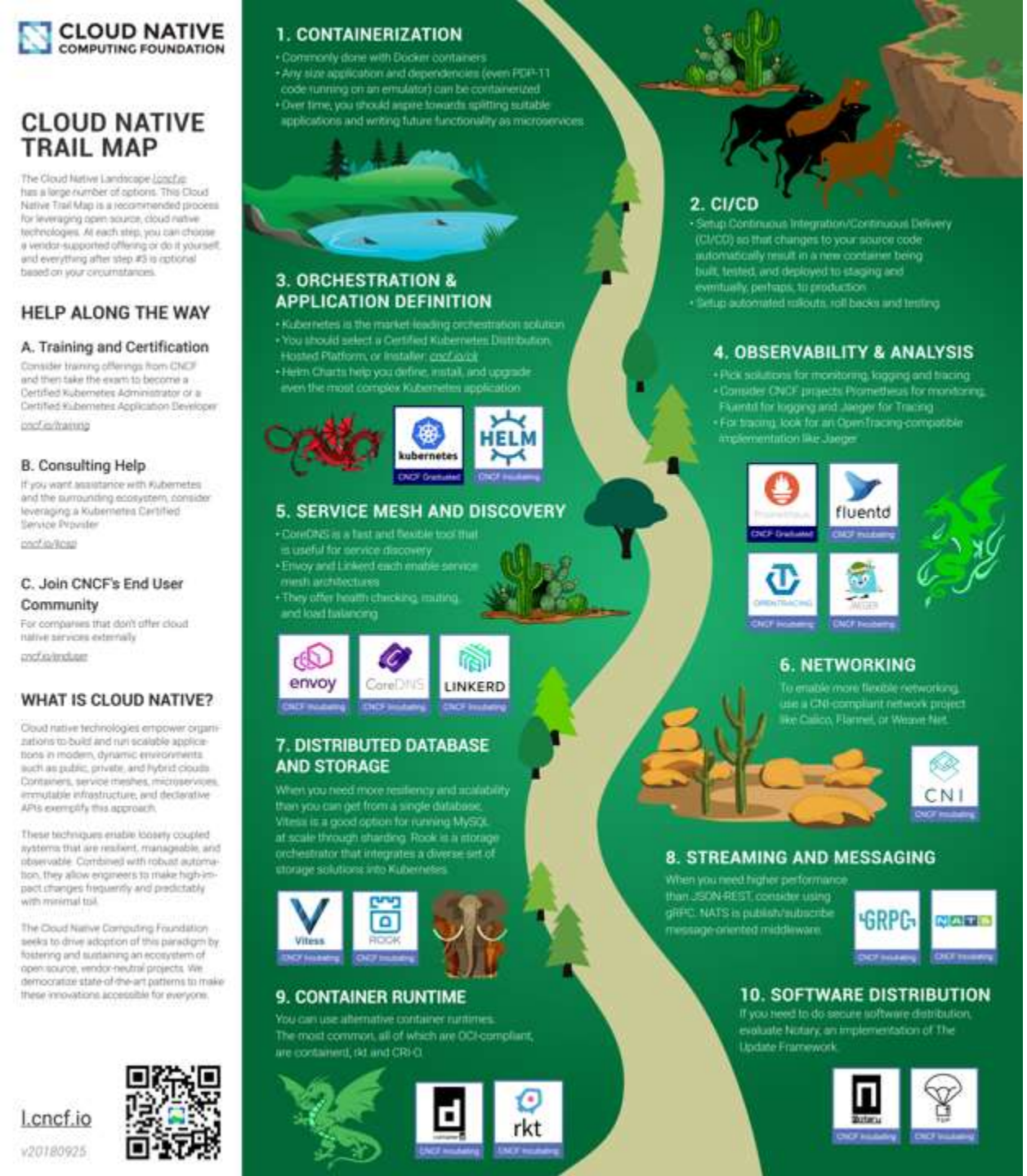
- Kubernetes provides a single unified platform to deploy containers across all your infrastructure

Develop new applications quickly

- Rich technical ecosystem.
- Large community
  - Kubernetes and CNCF slack has over 35k people
  - Plenty of meetups in many different cities
- Based on opensource
  - Developers can read the source code of platforms they are using
- Easier to find developers who want to work on newer technologies

Improve speed of innovation

- Cloud Native brings a new culture, technology and processes to accelerate innovation in organizations.
- Devops, CI/CD, Containerization modernizes your existing development teams
- Allows them to go much faster than before.

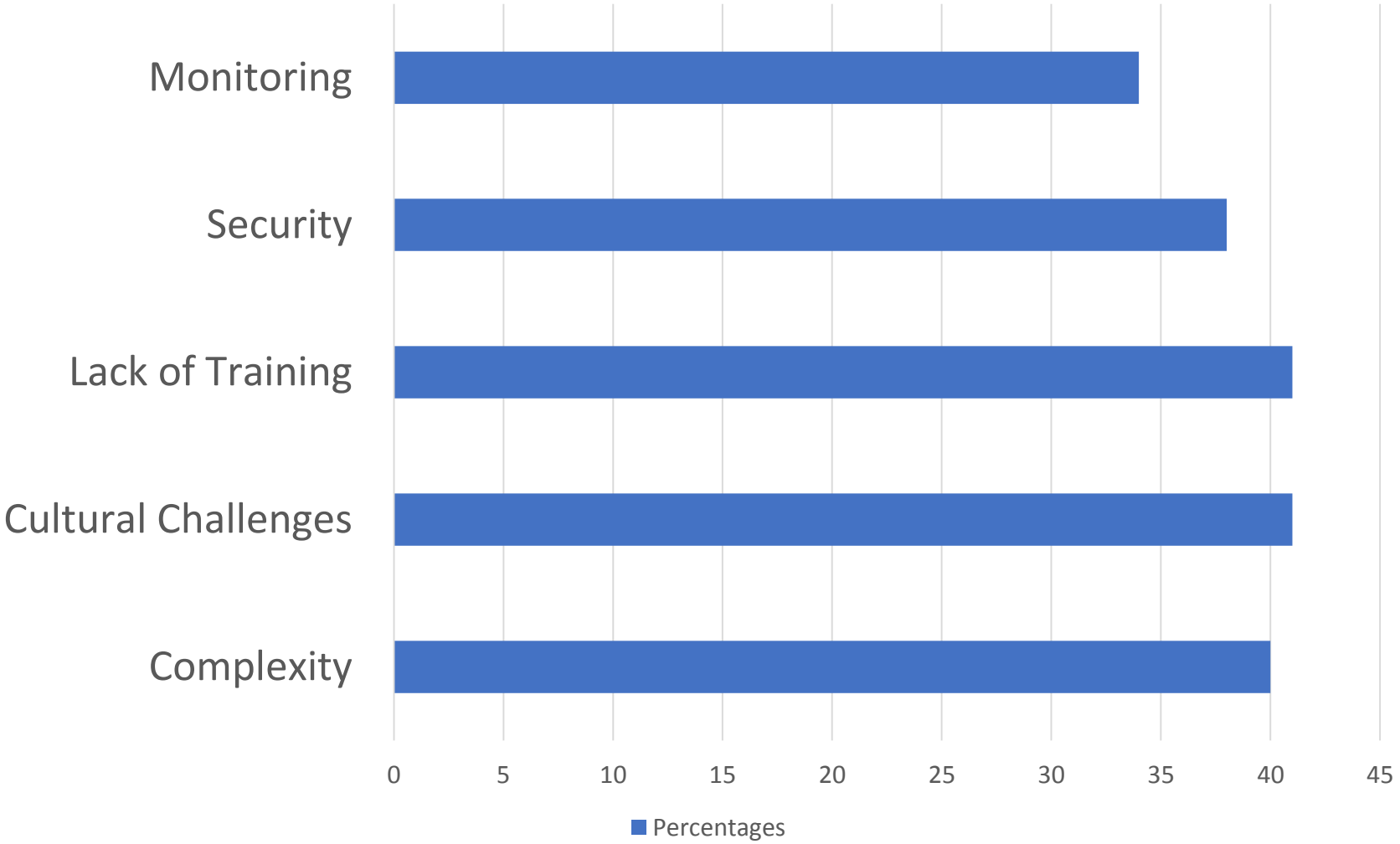


CNCF Trail Map

https://landscape.cncf.io/images/landscape.pdf

Challenges

Top 5 challenges to cloud native adoption…



https://www.cncf.io/blog/2018/08/29/cncf-survey-use-of-cloud-native-technologies-in-production-has-grown-over-200-percent/

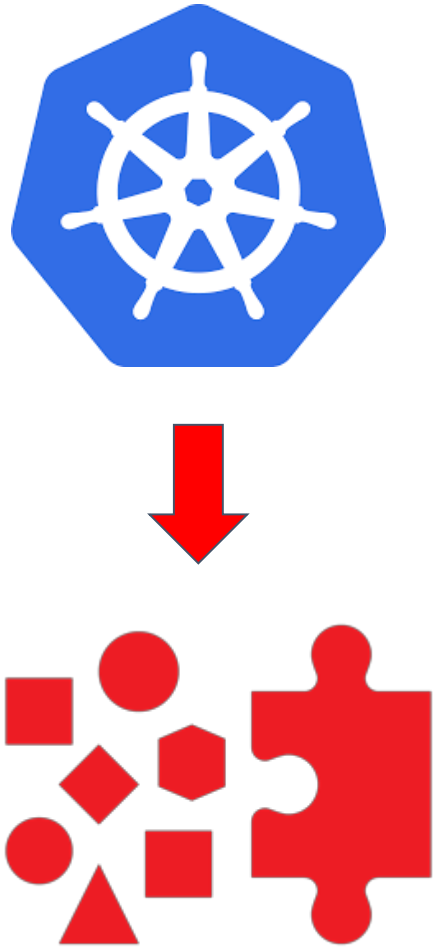
Other Challenges

- Storage (30% down from 41%)
- Networking (30% down from 38%)
- Reliability (17% down from 20%)
- Logging (25% down from 32%)
- Scaling (20% down from 24%)

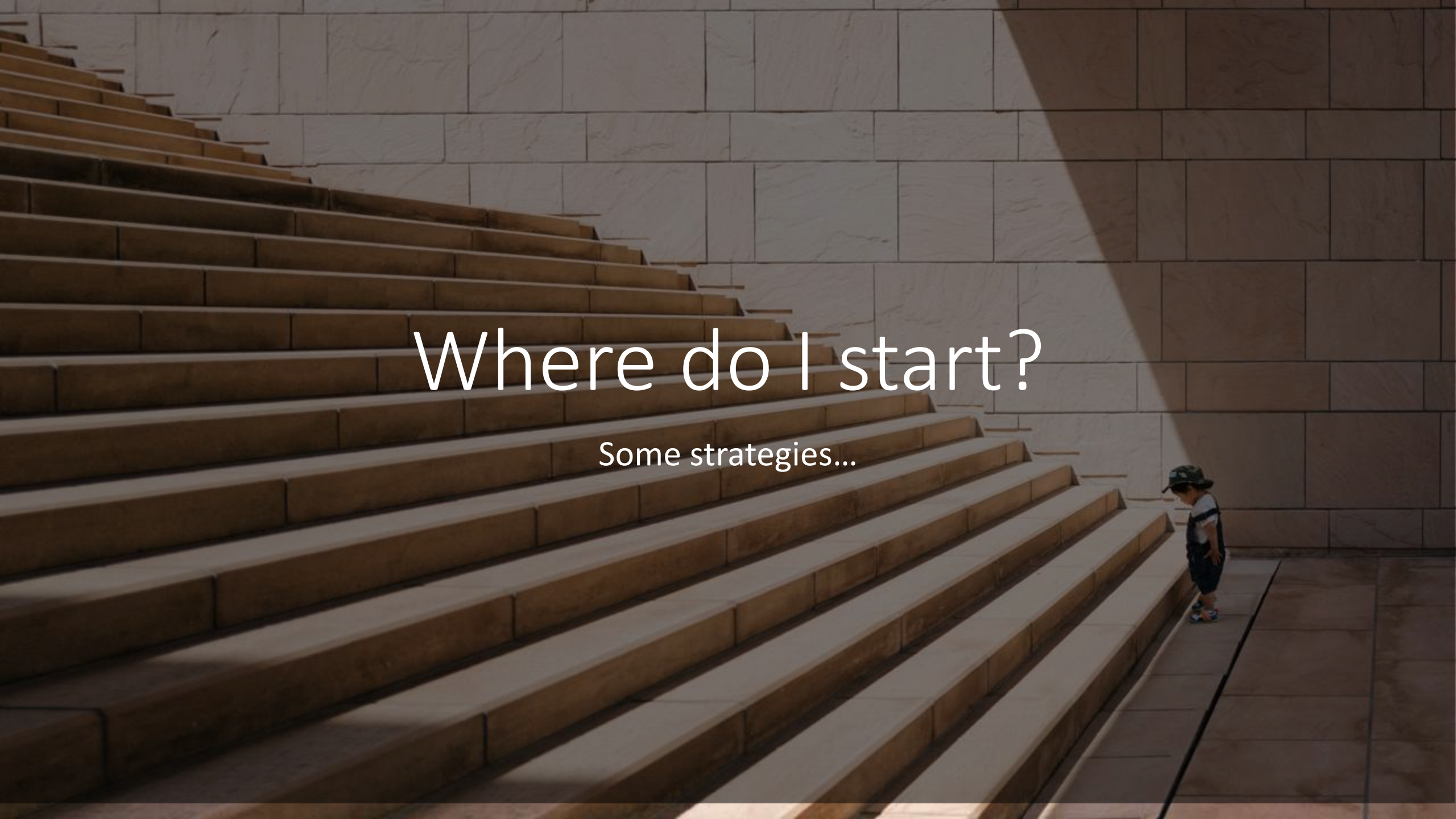
https://www.cncf.io/blog/2018/08/29/cncf-survey-use-of-cloud-native-technologies-in-production-has-grown-over-200-percent/

Kubernetes & Cloud Native Challenges

- Managing, maintaining, upgrading Kubernetes Control Plane
  - API Server, etcd, scheduler etc….
- Managing, maintaining, upgrading Kubernetes Data Plane
  - In place upgrades, deploy parallel cluster etc….
- Figuring out container networking & storage
  - Overlays, persistent storage etc… - it should just work
- Managing Teams
  - How do I manage & control team access to my clusters?
- Security, security, security



Source: Oracle Customer Survey 2018

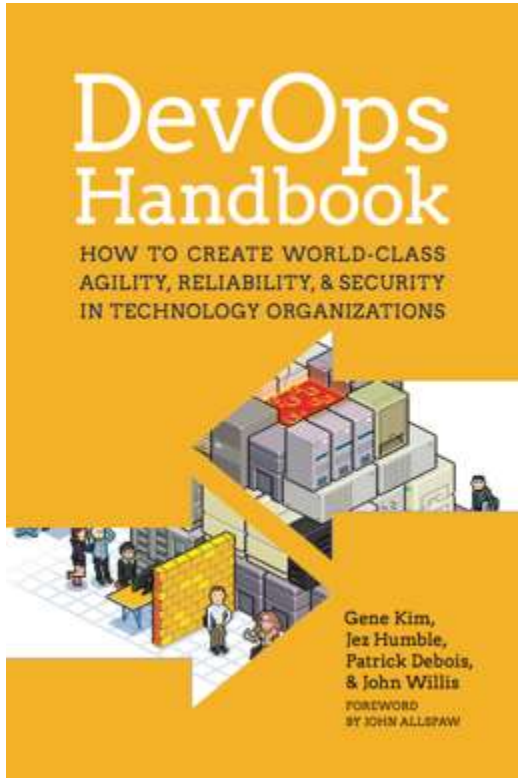


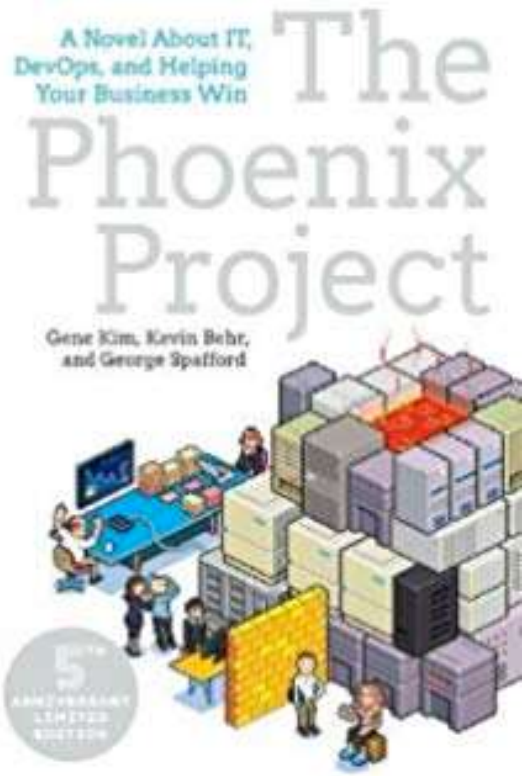
Where do I start?

Some strategies…

Silos

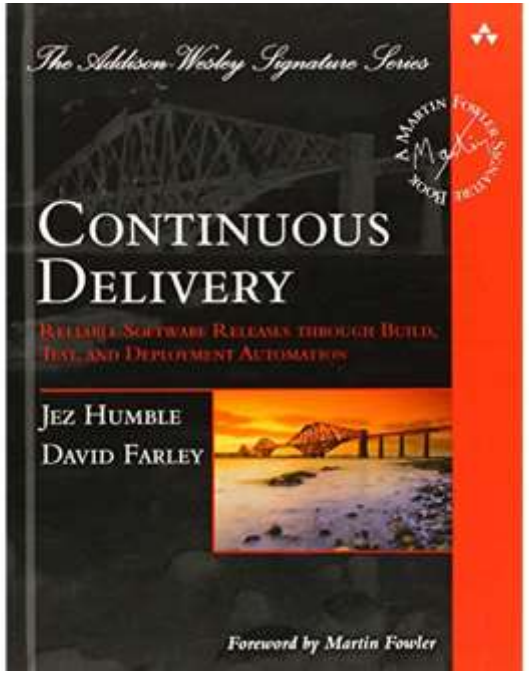
- Heavily siloed organizations can benefit from a devops mindset
- Use containers as a way to break down silos in your engineering orgs



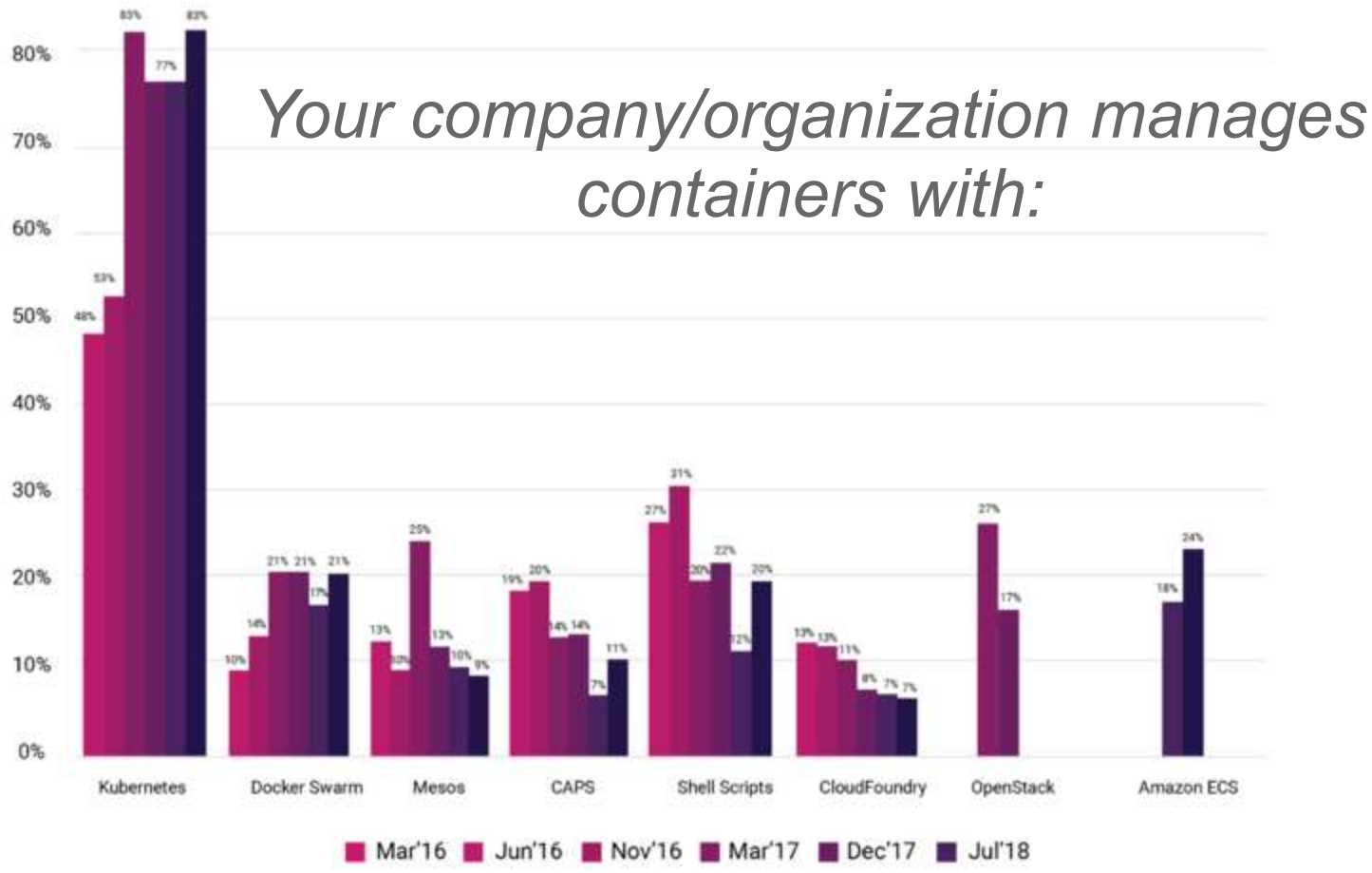


Releasing Code

- Step 1: Invest in Continuous Integration
- Step 2: Continuous Delivery



Orchestration?



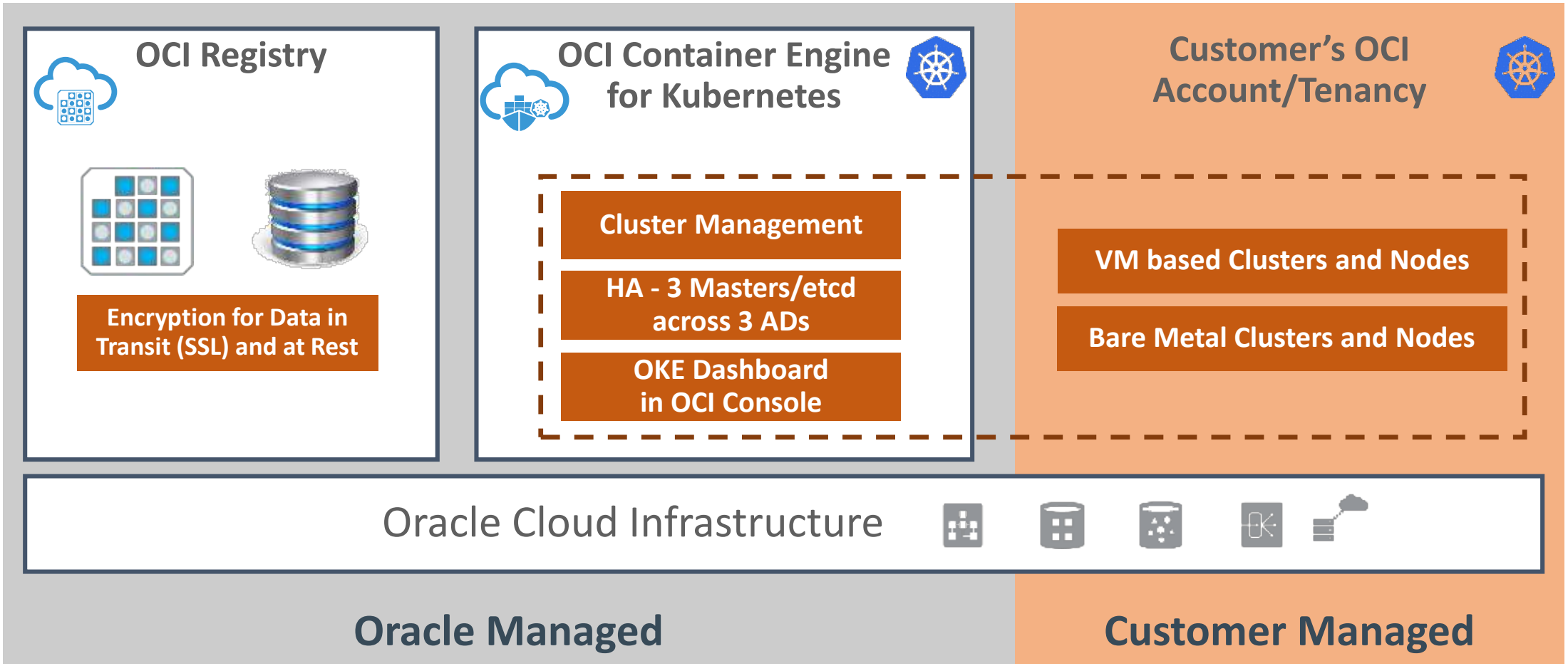

Nobody ever got
fired for ~~buying IBM.~~
using Kubernetes.

Biggest Lie: “Kubernetes is easy”

Kubernetes is complex

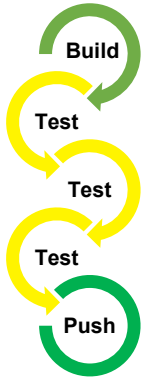
- Use a Kubernetes Managed Service
  - Like Oracle Container Engine for Kubernetes, Google Kubernetes Engine etc
- Benefits:
- Enables developers to get started and deploy containers quickly.
- Gives DevOps teams visibility and control for Kubernetes management.
- Combines production grade container orchestration of open Kubernetes, with control, security, IAM, and high predictable performance of cloud infrastructure
- Manage what you really need to manage

Kubernetes is Complex



End to End Workflow…

Any CI/CD - ie Jenkins, Oracle Pipelines, etc.

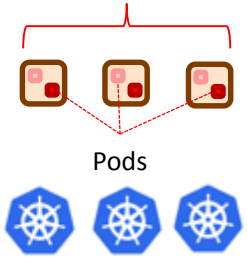


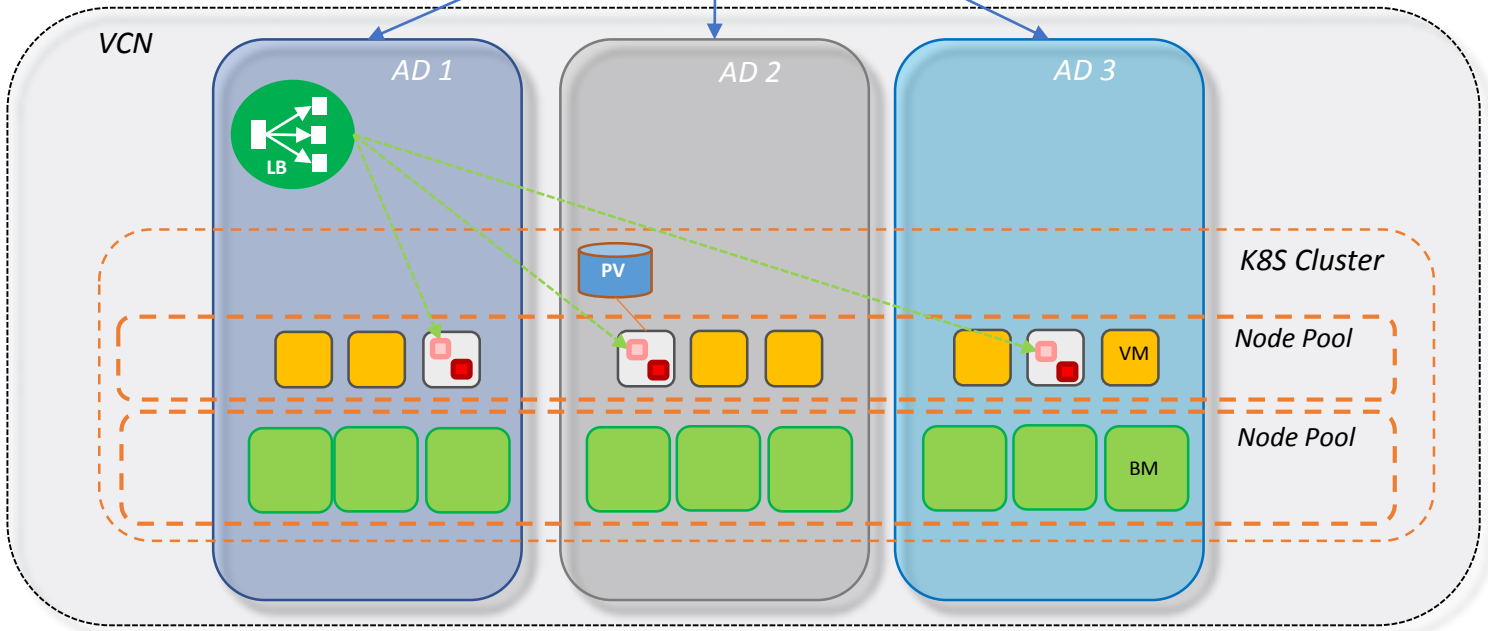
OCI Registry

OCI Container Engine for Kubernetes

- **Container Native**: Standard Upstream Kubernetes; Fully Managed Lifecycle; Integrated Registry
- **Developer Friendly**: Simple, Streamlined User Interface; REST API; Helm, and DNS Built-in
- **Enterprise Ready**: Oracle Cloud Infrastructure Performance; Highly Available; Secure with OCI Access Controls

Exposed Kubernetes Service





Take It slow…

Greenfield OR Strangler application patterns