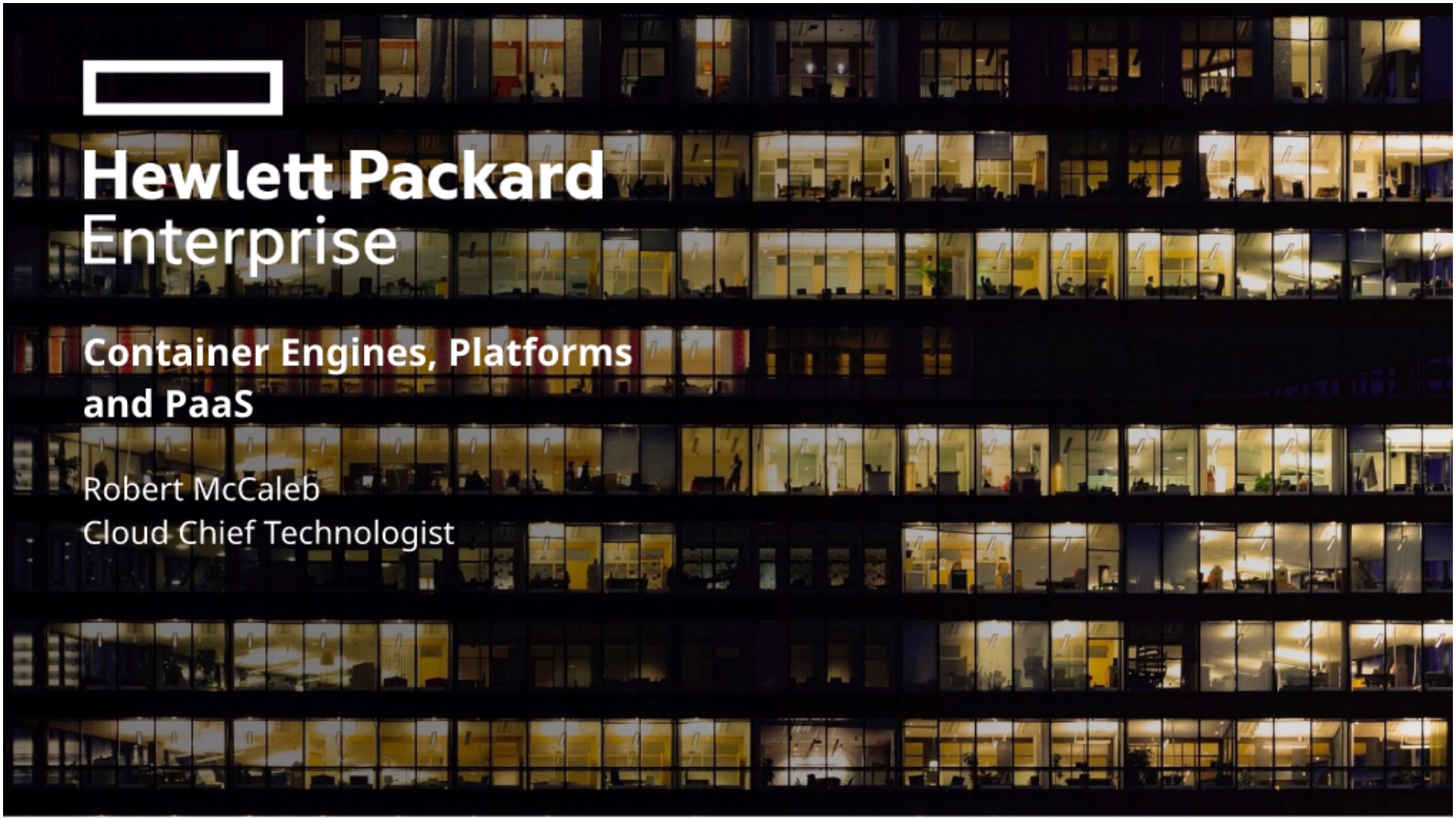
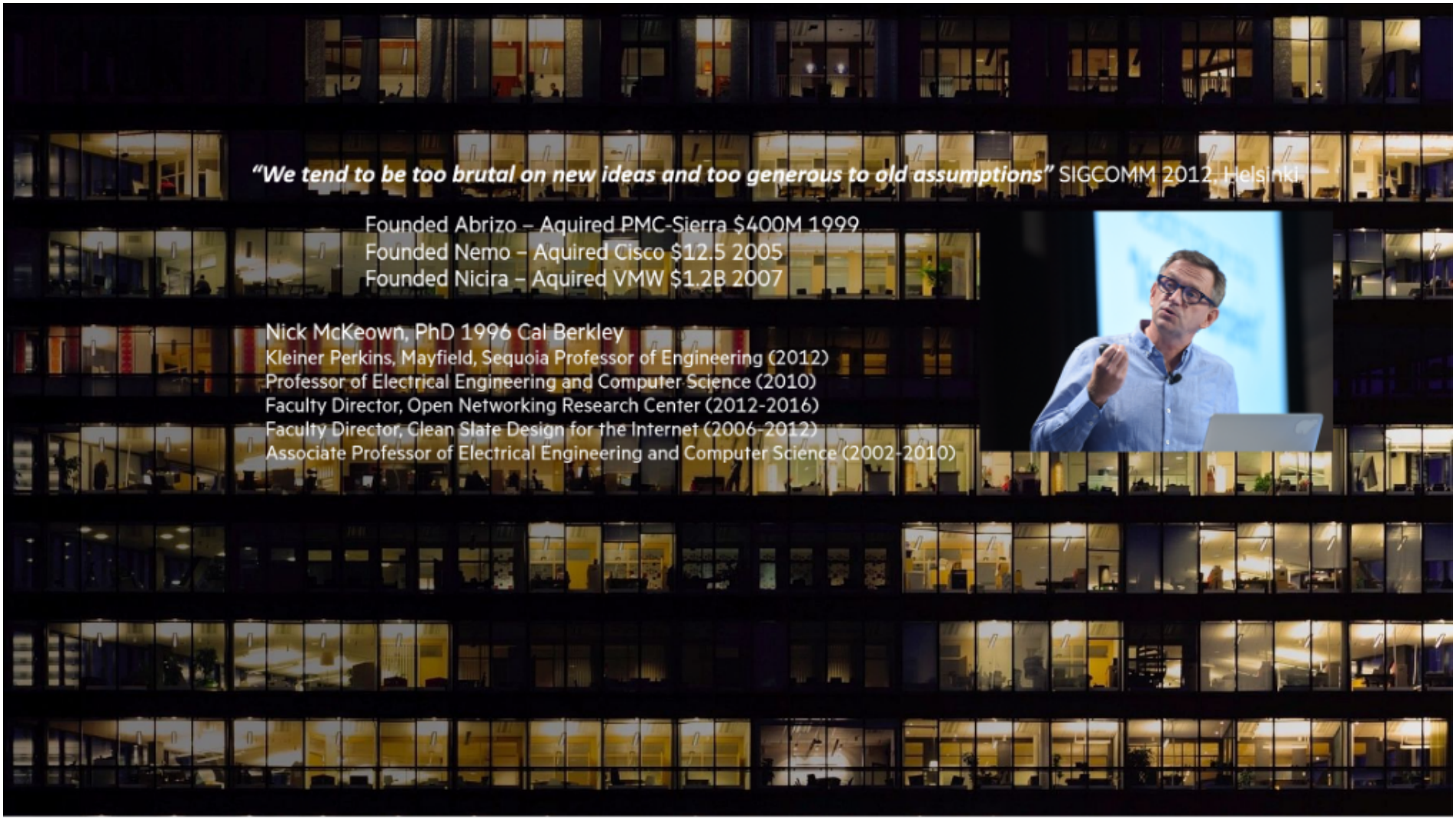
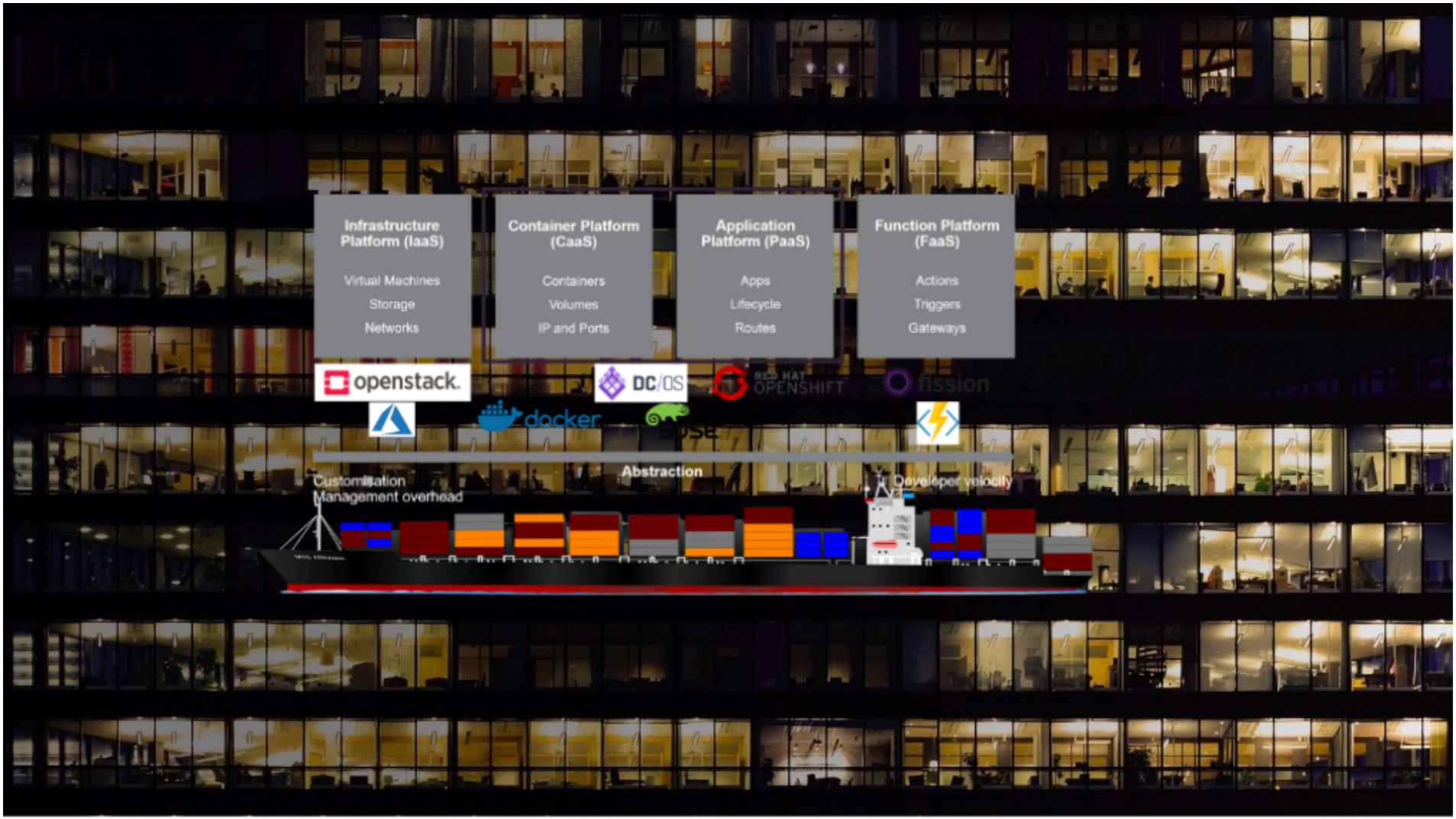

Hewlett Packard
Enterprise
Container Engines, Platforms
and PaaS
Robert McCaleb
Cloud Chief Technologist


"We tend to be too brutal on new ideas and too generous to old assumptions" SIGCOMM 2012, Helsinki
Founded Abrizo – Aquired PMC-Sierra $400M 1999
Founded Nemo – Aquired Cisco $12.5 2005
Founded Nicira – Aquired VMW $1.2B 2007
Nick McKeown, PhD 1996 Cal Berkley
Kleiner Perkins, Mayfield, Sequoia Professor of Engineering (2012)
Professor of Electrical Engineering and Computer Science (2010)
Faculty Director, Open Networking Research Center (2012-2016)
Faculty Director, Clean Slate Design for the Internet (2006-2012)
Associate Professor of Electrical Engineering and Computer Science (2002-2010)

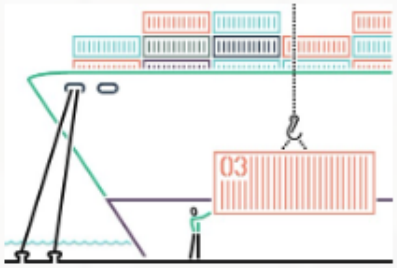

VS


Infrastructure Platform (IaaS)
Virtual Machines
Storage
Networks
Container Platform (CaaS)
Containers
Volumes
IP and Ports
Application Platform (PaaS)
Apps
Lifecycle
Routes
Function Platform (FaaS)
Actions
Triggers
Gateways
openstack.
DC/OS
RED HAT OPENSHIFT
fission
docker
Abstraction
Customisation
Management overhead
Developer velocity


Big Picture
PaaS
Frameworks
Container

Containers

Components



Minimum Viable Container

Containers in the Enterprise

Containers Ain't VM's

Example

Minimum Viable Container

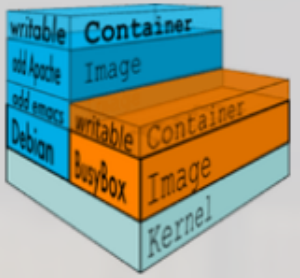
X86 HW

Container Engine

containerd

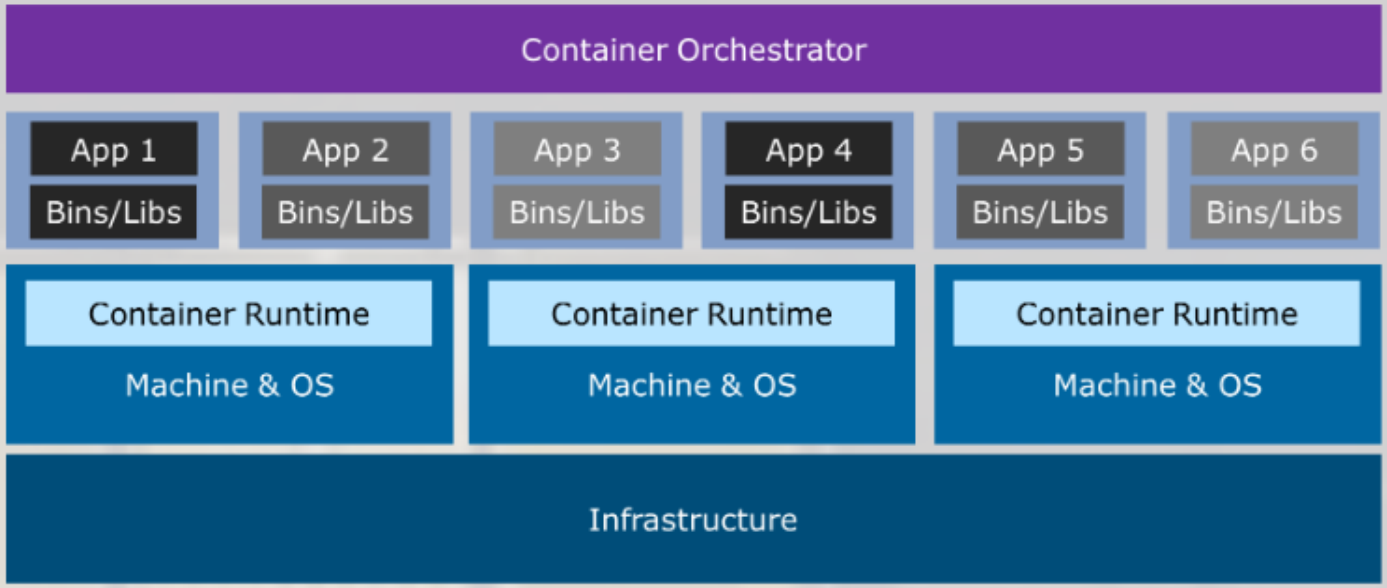
Storage

Container



OS

Layered Look

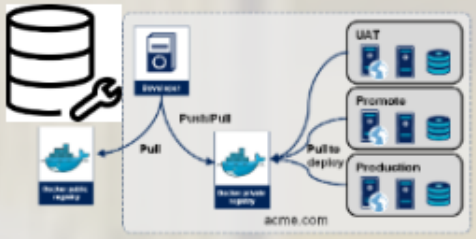


Containers in the Enterprise

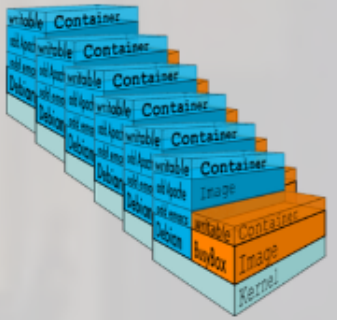
HW Scale

Cluster

Network



Container

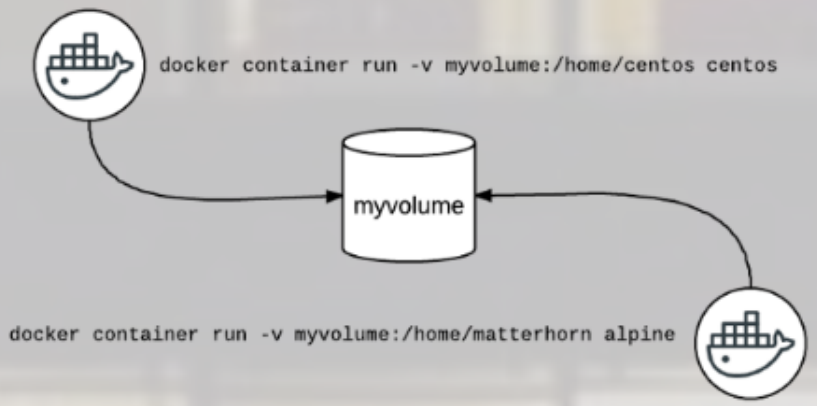


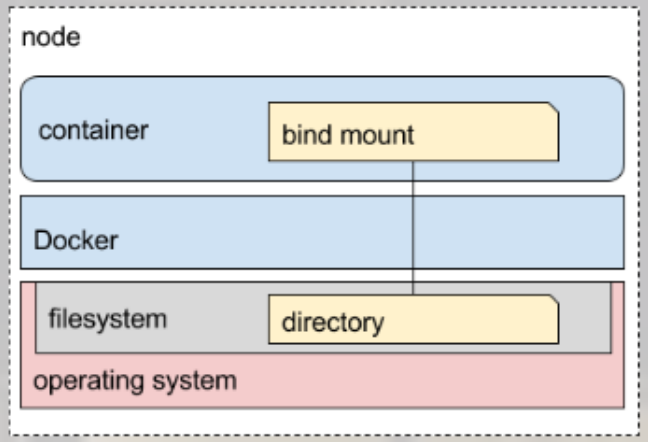
OS

Containers Ain't VM's

How do I back up my container?

You don't, the container is stateless. No unique data actually lives in the container. Persistent data lives in an named volume. That volume is backed up.

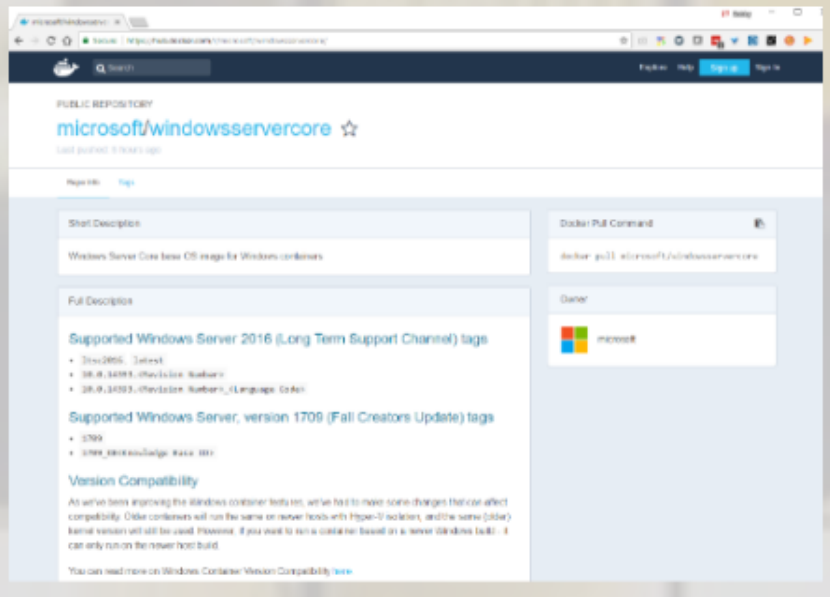




Patching?

What's my patch management strategy for my running containers? Where does the application run?

Containers are built on a "rip and replace" premiss. You always have the latest version. The App is installed at the time of build




```
FROM microsoft/windowsservercore:latest

SHELL ["powershell", "-Command", "$ErrorActionPreference = 'Stop';
$ProgressPreference = 'SilentlyContinue';"]

RUN Add-WindowsFeature Web-Server; `
    Add-WindowsFeature NET-Framework-45-ASPNET; `
    Add-WindowsFeature Web-Asp-Net45; `
    Remove-Item -Recurse C:\inetpub\wwwroot\*; `
COPY C:\codeDIR C:\inetpub\wwwroot

ENV ROSLYN_COMPILER_LOCATION c:\\RoslynCompilers\\tools

EXPOSE 80
```

Example


$5 per month
Apps
System Services
Container Infrastructure Orchestration
Container Runtime
Image Store (Registry)
Infrastructure (virtual/physical, OS)
ghost
ghost
ghost
NGINX
containerd
DigitalOcean


Big Picture
PaaS
Frameworks
Container

Frameworks

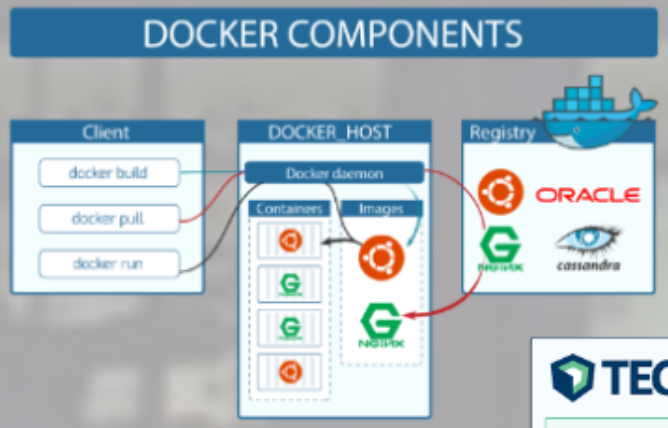
Setting the Foundation for CaaS

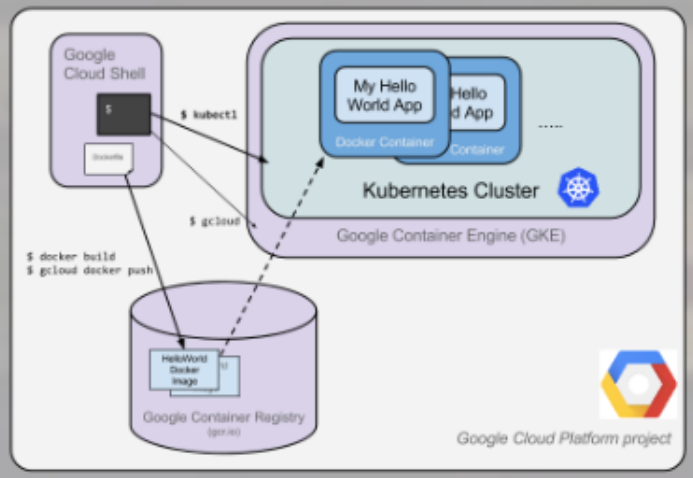
What is considered a framework

Major components of a framework

Outside the framework

What is considered a framework





What is considered a framework

Container Infrastructure Orchestration

Container Runtime

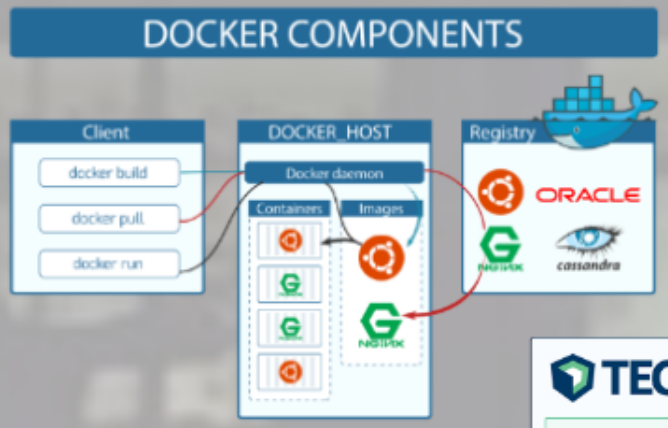
Image Store (Registry)

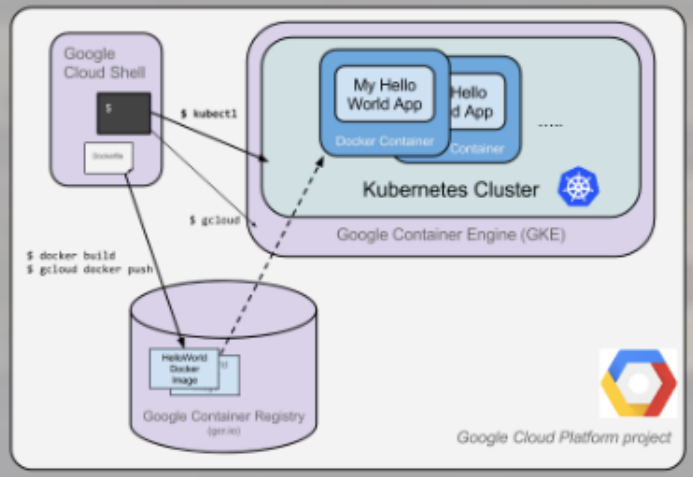
Infrastructure (virtual/physical, OS)

Scheduling, Resource Mgt, Service Mgt

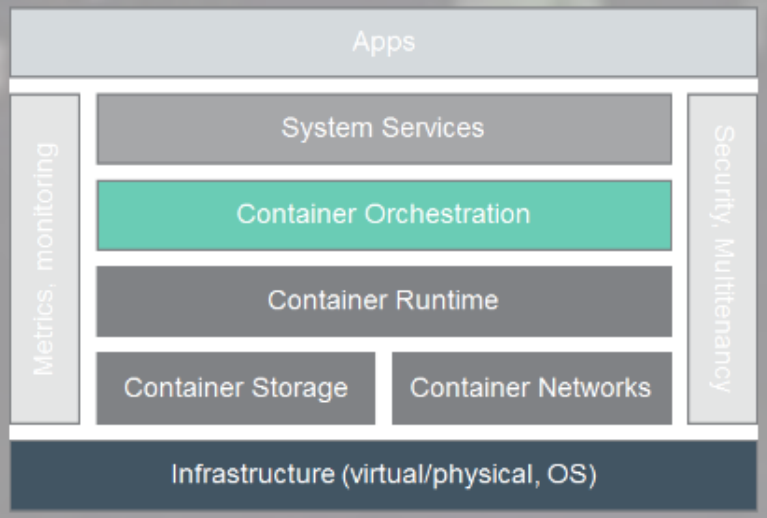
containerd

What is considered a framework





Major components of a framework



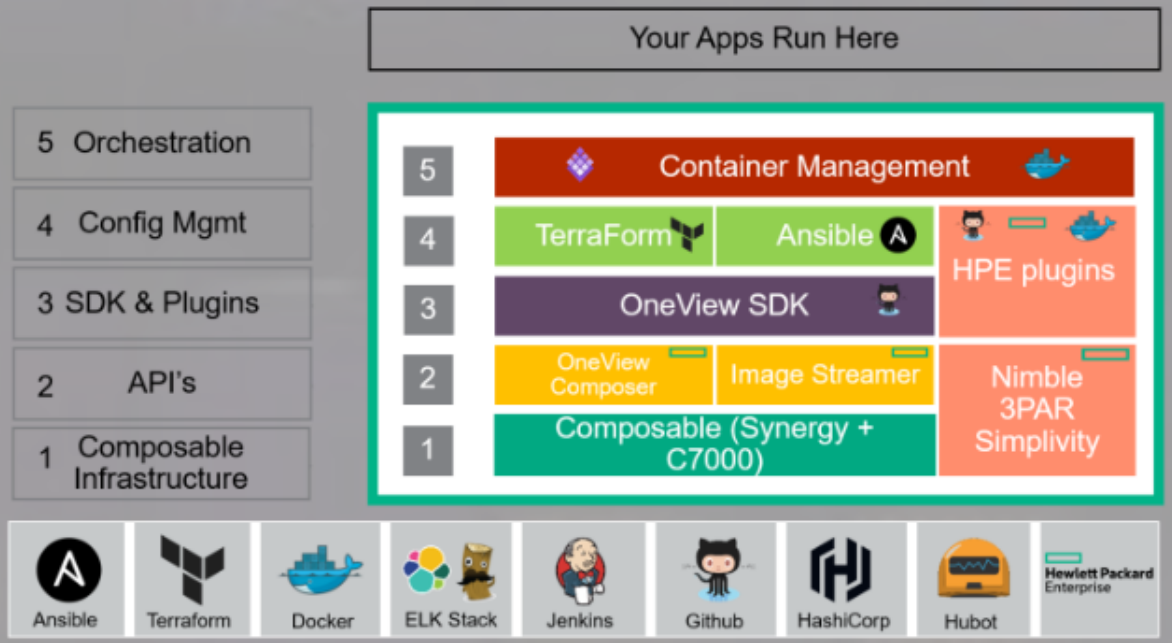
Scheduling
Task placement, replication/scaling, readiness checks, resurrection/rescheduling, rolling updates, co-location

Resource management
Memory, CPU, GPU, storage capacity, ephemeral volumes, remote/local persistent volumes, ports, IP

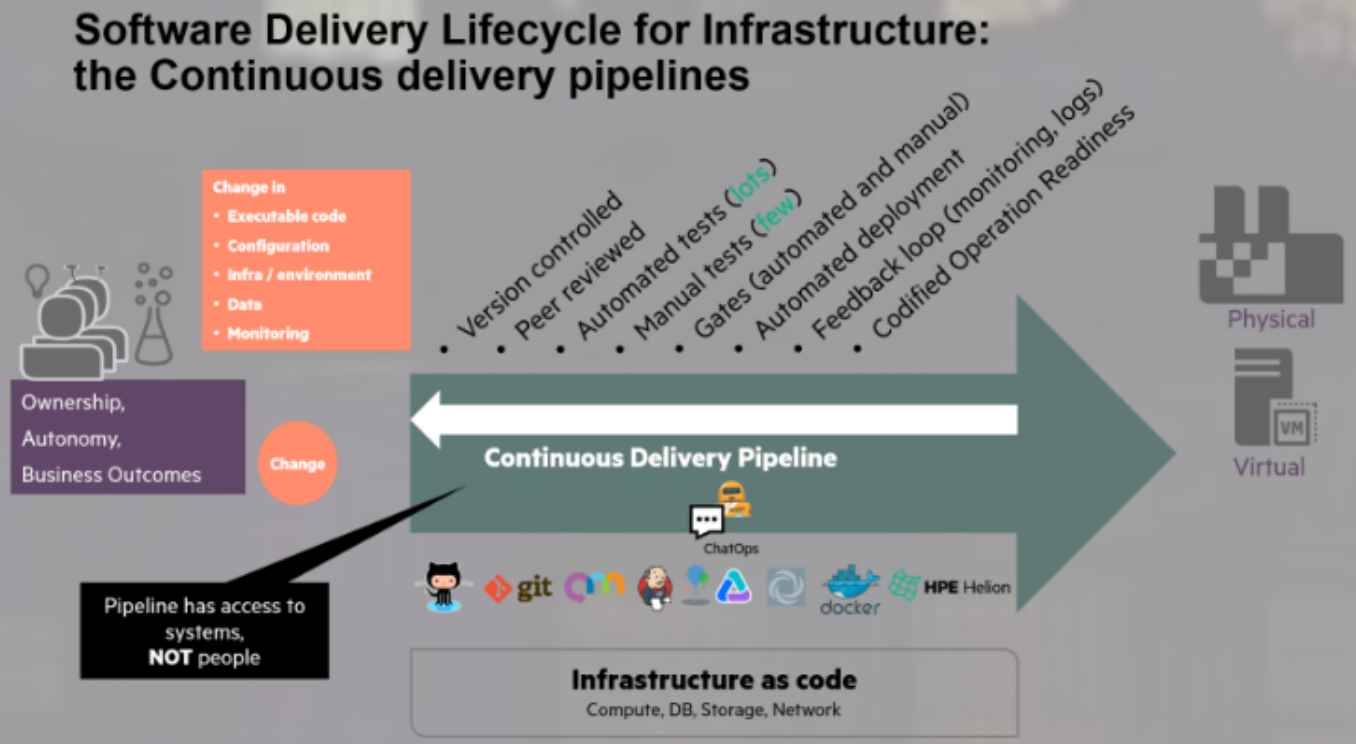
Service management
Labels, groups/namespaces, dependencies, LB, VIPs, DNS, secrets, configuration management

HPEIT CaaS framework v1

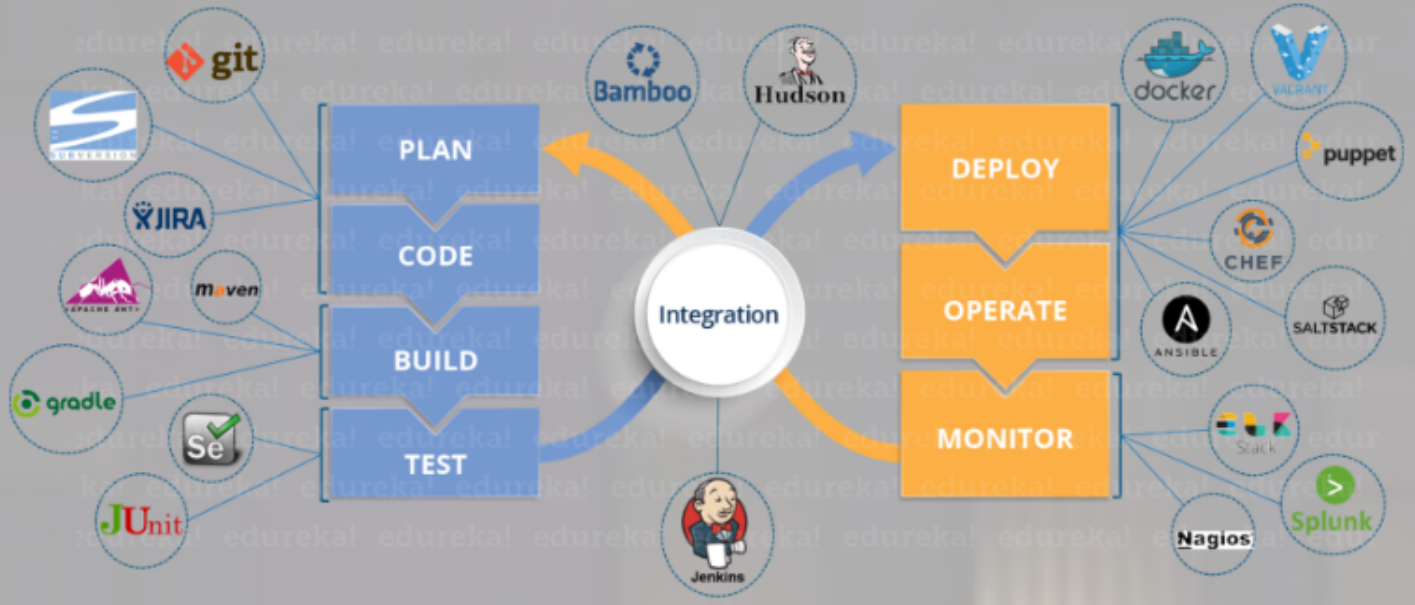
Container as a Service building blocks



HPEIT CaaS framework v1



Outside the framework




Big Picture
PaaS
Frameworks
Container

PaaS

The On Ramp to Automation (CI/CD)

What is considered a PaaS

MesoSphere

Cloud Foundry

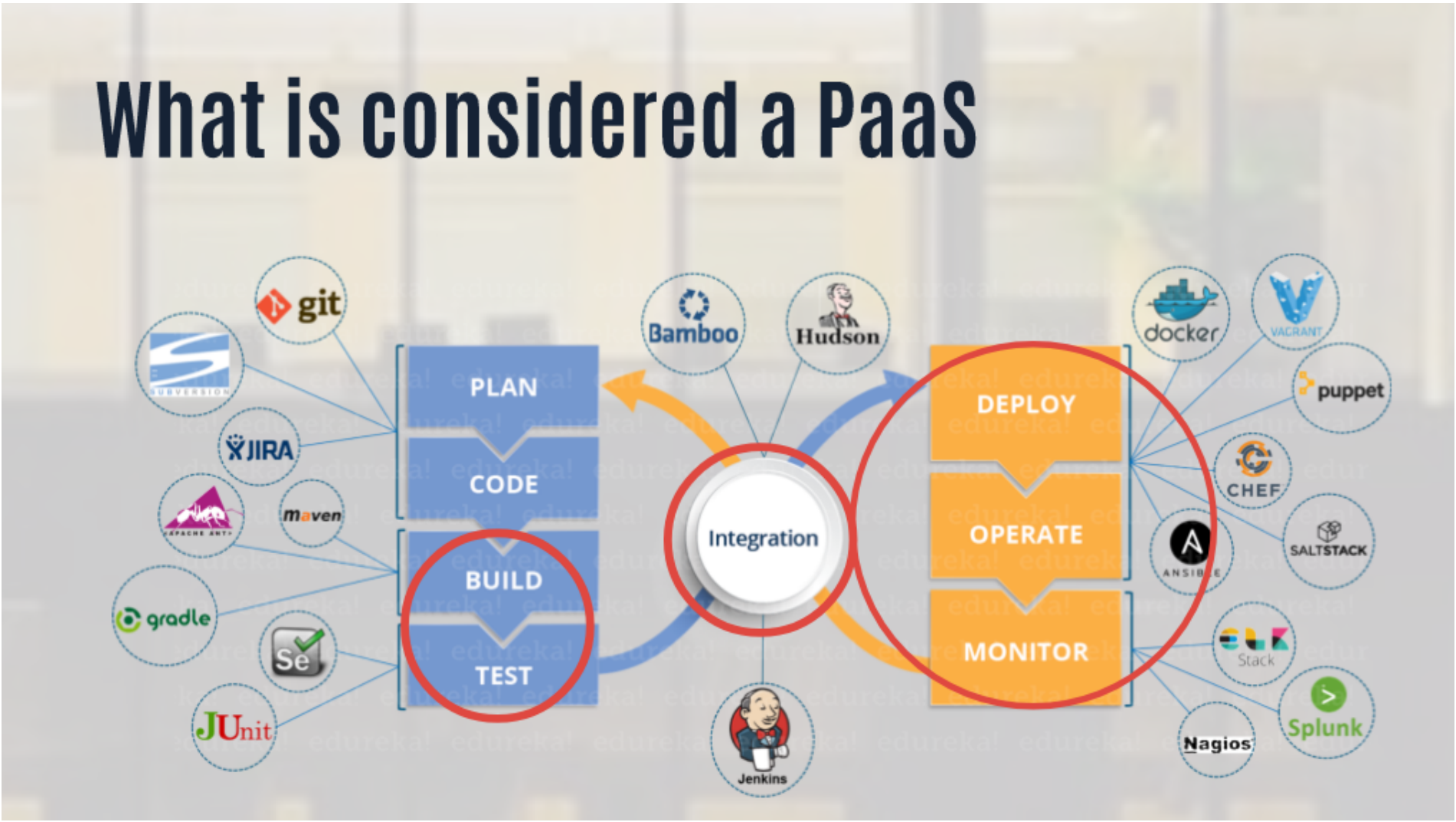
OpenShift

What is considered a PaaS

Provides a **platform** allowing customers to **develop, run, and manage applications without** the complexity of **building** and **maintaining** the **infrastructure** typically associated

PaaS includes infrastructure—servers, storage, and networking—**but also middleware, development tools, business intelligence (BI) services, database management systems, and more.** PaaS is designed to support the complete **web application lifecycle**: building, testing, deploying, managing, and updating.

What is considered a PaaS



The Big 3

Difference of Mindsets

CLOUDFOUNDRY

12 Factor Preferred (Only)

OPENSHIFT

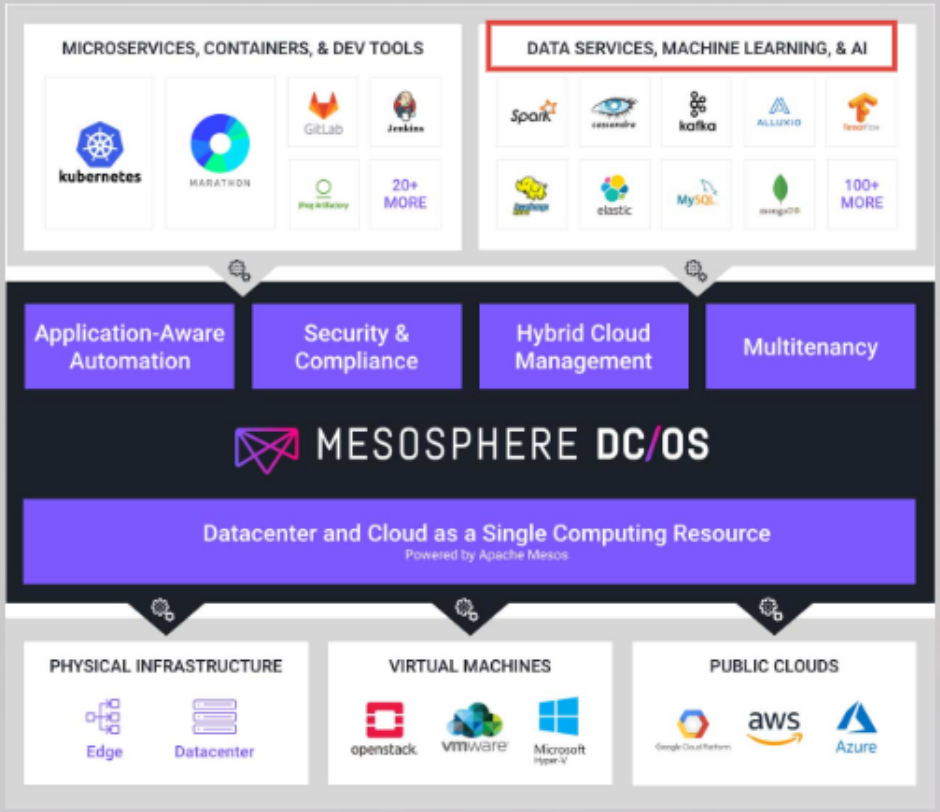
State-full Friendly

MESOSPHERE

Complex Middle-ware


MESOSPHERE

Is and Isn't a PaaS



An AWS like experience to our users. Data Scientists

Deploy, Manage and Maintain Complex platforms for consumption

More of a Framework for Frameworks then a PaaS or CaaS

Manages the underlying infrastructure as well

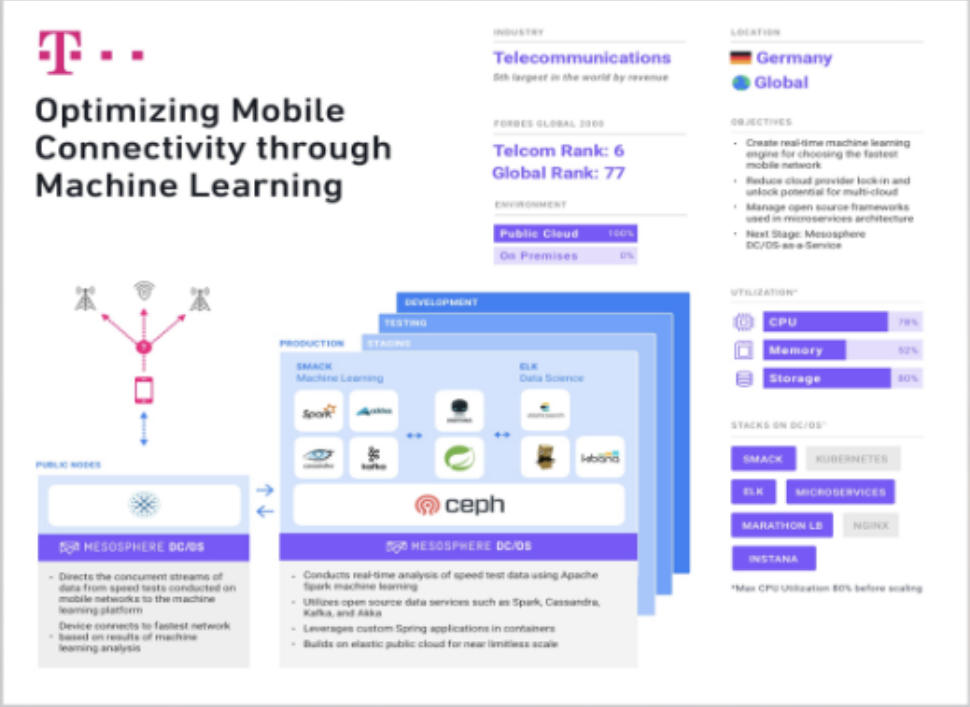
Fast Data versus Big Data

- Apache **Spark** for large-scale analytics
- Apache **Mesos** for running data services and containers elastically
- **Akka** for simplifying development of data-driven apps
- Apache **Cassandra** for highly available and scalable storage
- Apache **Kafka** for capturing message streams

Fast Data versus Big Data

Historical data ingested for “Big Data” analytics.

Fast Data versus Big Data



Connect App allows customers to choose between hotspots and mobile data based on cost and performance and data type.

Mesosphere

The upside and the downside

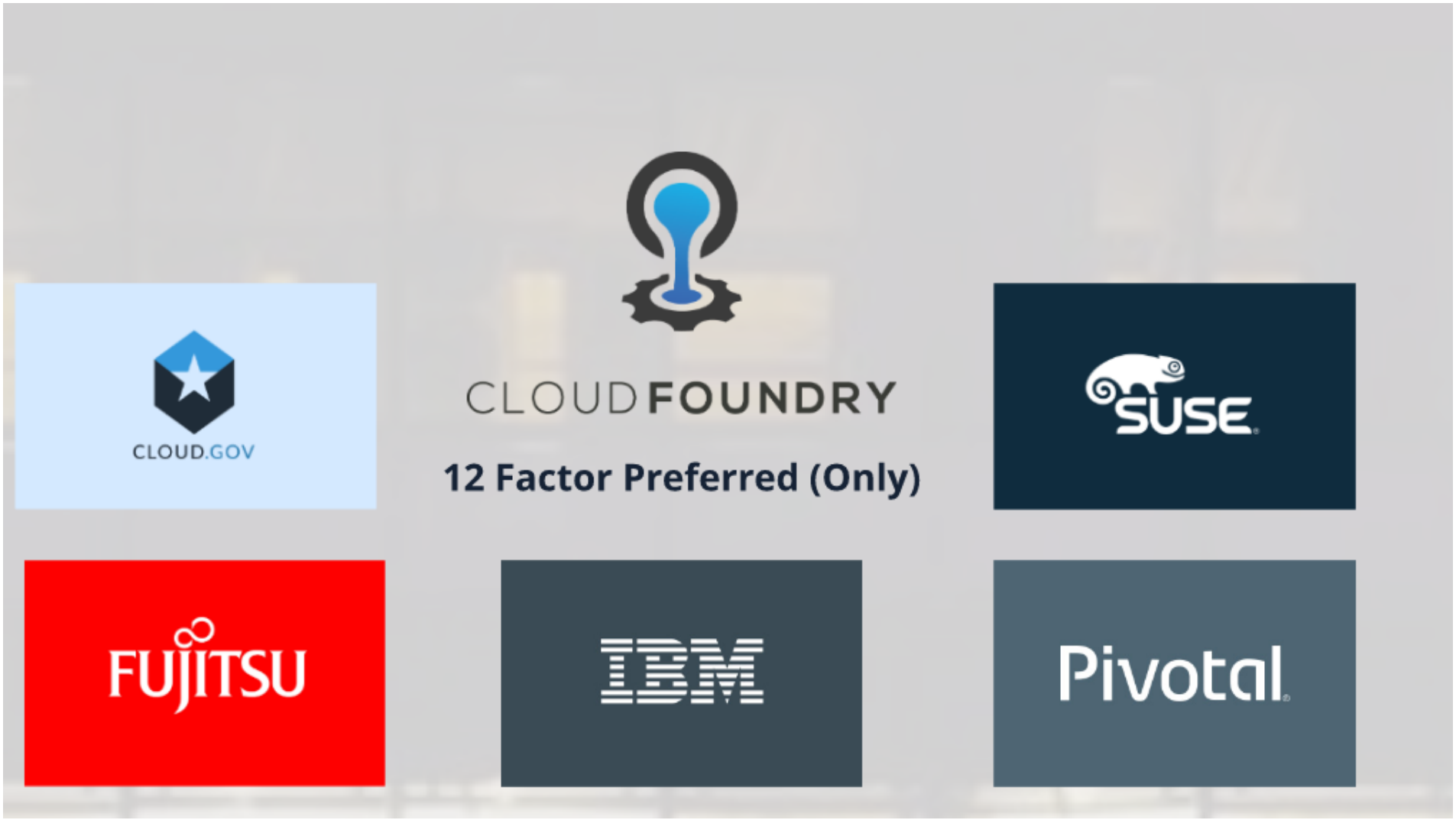
Upside

- Makes complex application frameworks really easy... (SMACK)
- Manages not only the application but also the infrastructure
- True Hybrid approach

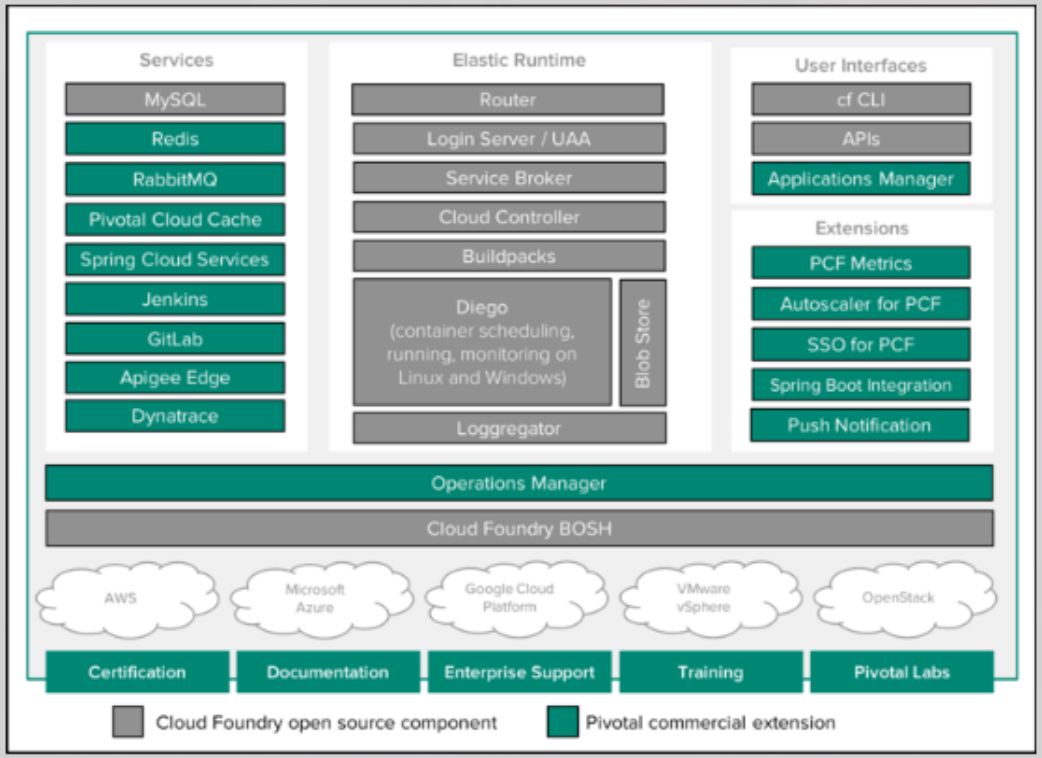
Downside

- Negative perception of Zookeeper
- Smaller Eco System (catalog)
- More appealing to Data Scientists then general cloud technologist


CLOUDFOUNDRY
12 Factor Preferred (Only)


CLOUD.GOV
CLOUDFOUNDRY
12 Factor Preferred (Only)
SUSE
FUJITSU
IBM
Pivotal

CF Open and Pivotal Enterprise



Cloud Foundry open source is open for everyone

Pivotal makes it Enterprise ready

12 Factor

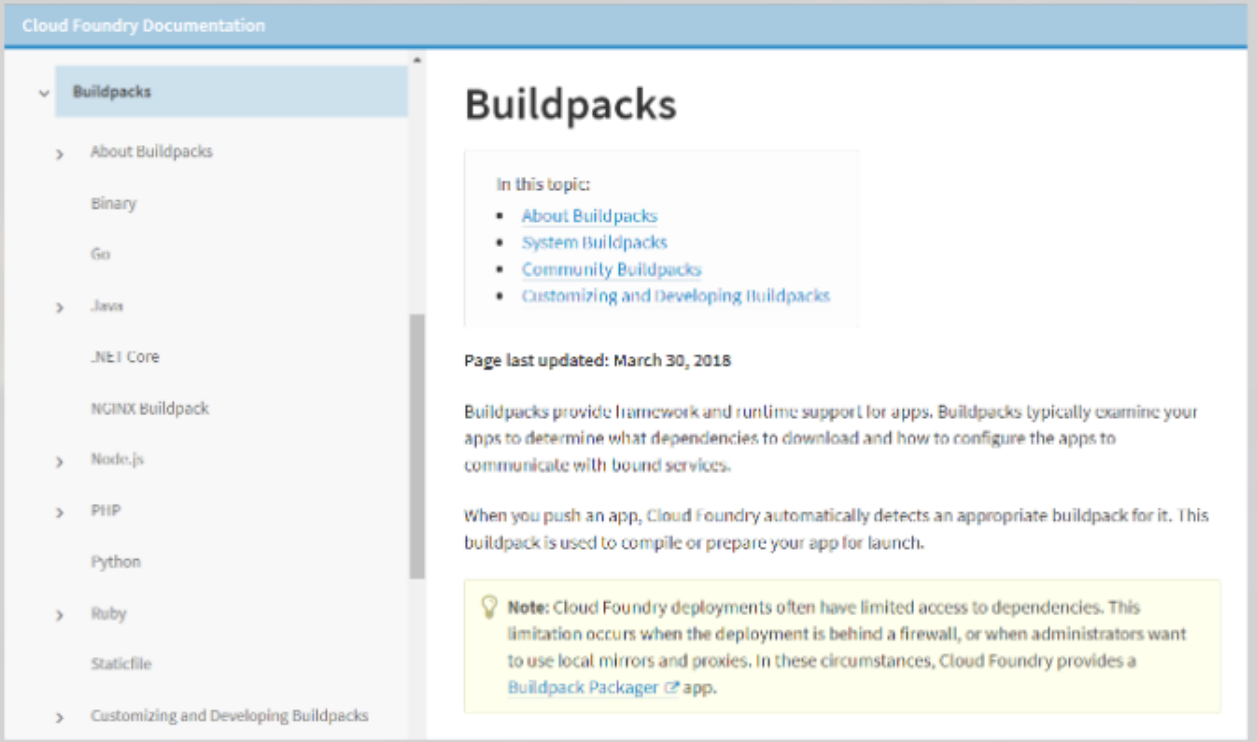
Cloud Foundry

12 Factor Stateless

BuildPacks
Key Tenant for abstracting dependencies

Droplet
Built app code+deps

App is Live
Traffic, QOS, Scaling



CF

The upside and the downside

Upside

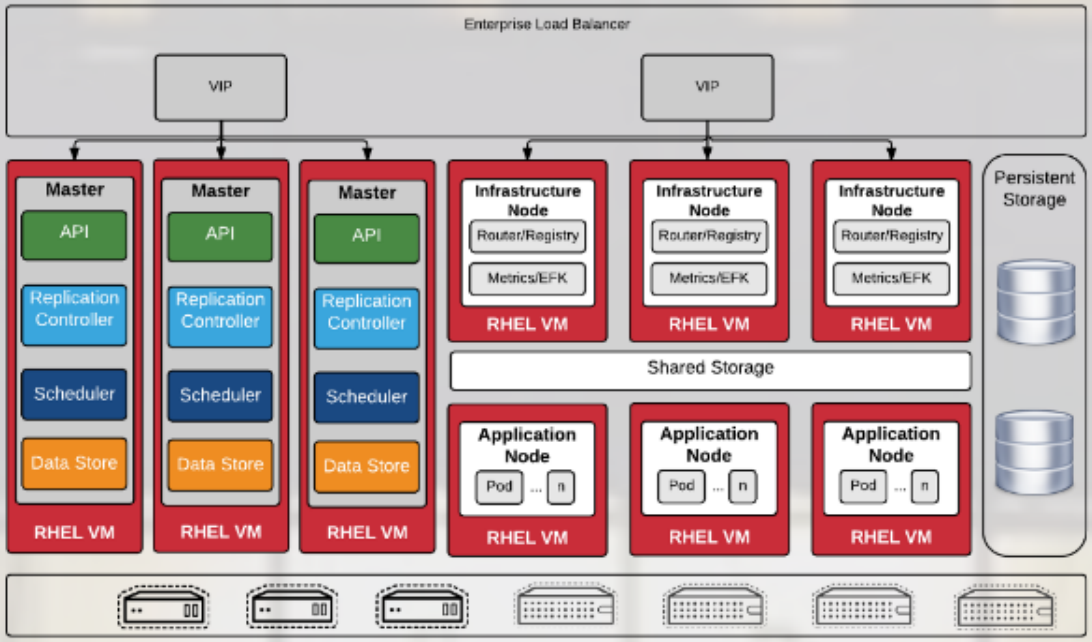
- Extremely fast to get developers and apps up and running. HPE PeopleFinder 3 months
- Forces true 12 factor app principles
- Stateless +++
- The infrastructure implementation is average in complexity

Downside

- Pure Stateless. State-full data (DB) need to be outside or a "service"
- Difficult (at best) to lift and shift apps to
- Most legacy apps require complete re-factoring


OPENSHIFT

Hello Enterprise



Complete PaaS for more traditional apps

J2EE sweet spot (JBoss)

Persistent data friendly

Kubernetes from day 1

CoreOS acquisition (Quay)

OpenShift

The upside and the downside

Upside

- Trusted history with RedHat
- Persistent data
- Legacy Apps Freindly

Downside

- Lots of physical infrastructure
- Difficult to implement
- "Source to Image" takes too long
- Forked versions of Docker and other
- Windows????

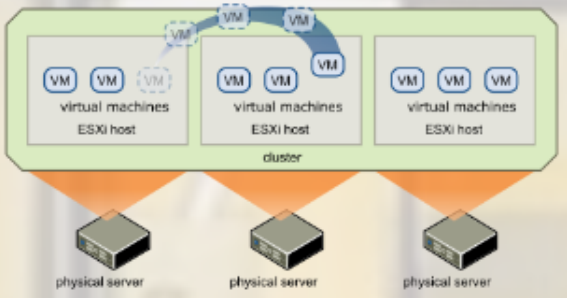

Big Picture
PaaS
Frameworks
Container

Big Picture

Which One is Right


Which One is Right
MESOSPHERE
CLOUDFOUNDRY
OPENSHIFT
DECISION
RIGHT
WRONG
MESOSPHERE
CLOUDFOUNDRY
OPENSHIFT

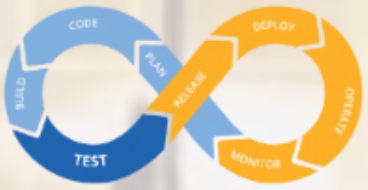
Choices, Guardrails and Maturing



serverless

12-18 Months

Process is the Key



Software Delivery Lifecycle for Infrastructure: the Continuous delivery pipelines

Ownership,
Autonomy,
Business Outcomes

Change in
- Executable code
- Configuration
- Infra / environment
- Data
- Monitoring

Change

Pipeline has access to systems, **NOT** people

- Version controlled
- Peer reviewed
- Automated tests (lots)
- Manual tests (few)
- Gates (automated and manual)
- Automated deployment
- Feedback loop (monitoring, logs)
- Codified Operation Readiness

Continuous Delivery Pipeline

ChatOps

git docker HPE Helion

Infrastructure as code
Compute, DB, Storage, Network

Physical

VM
Virtual


Big Picture
PaaS
Frameworks
Container