

Intro To Containers, Container Storage and Docker

Alex McDonald, NetApp (Moderator)
Keith Hudgins, Docker
Chad Thibodeau, Veritas Technologies

October 6, 2016

SNIA Presenters



Chad Thibodeau
Veritas Technologies



Keith Hudgins
Docker



Alex McDonald
Chair SNIA Cloud
Storage Initiative

- The material contained in this presentation is copyrighted by the SNIA unless otherwise noted.
- Member companies and individual members may use this material in presentations and literature under the following conditions:
 - ◆ Any slide or slides used must be reproduced in their entirety without modification
 - ◆ The SNIA must be acknowledged as the source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA.
- Neither the author nor the presenter is an attorney and nothing in this presentation is intended to be, or should be construed as legal advice or an opinion of counsel. If you need legal advice or a legal opinion please contact your attorney.
- The information presented herein represents the author's personal opinion and current understanding of the relevant issues involved. The author, the presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

Agenda

1.Intro to Containers
and how they enable
DevOps & CI-CD

2.What is Docker?

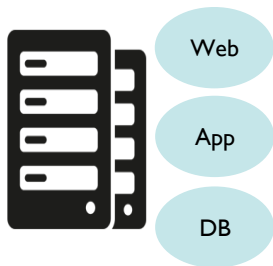
3.Persistent Storage
for Containers

4.Docker Data Center

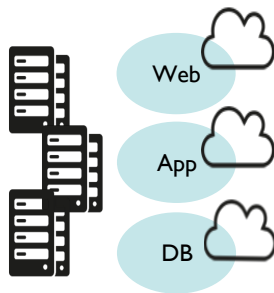
5.Docker Case
Studies

Application Deployment History

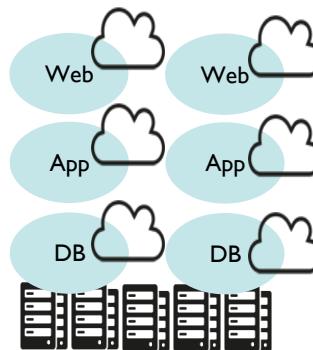
Monolithic Apps on Physical



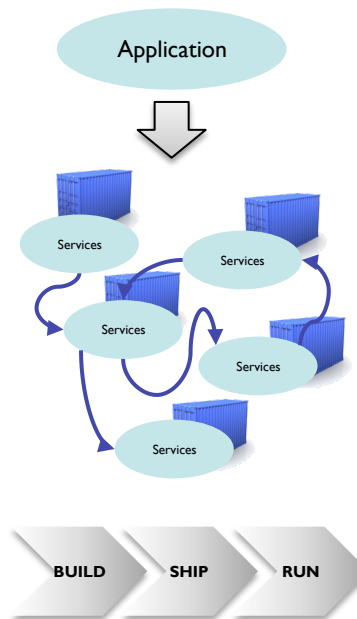
Virtual Machine Abstraction



Stateless & Horizontal Scalable Apps



Micro-services & Containers



Intro to Containers and how they enable DevOps & CI-CD

Poll Question #1

Are You Currently Using Containers in?

- A. Test and Dev only
- B. Test and Dev and Production
- C. Neither

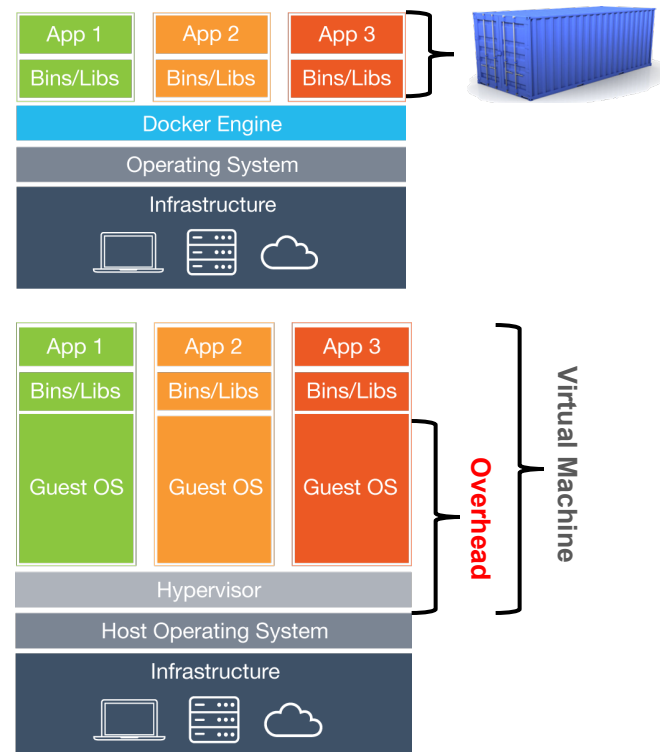
Containers 101

Containers

- Virtualization of application instead of hardware
- Runs on top of the core OS (Linux or Windows)
- Doesn't require dedicated CPU, Memory, Network—managed by core OS
- Optimizes Infrastructure—speed and density

“**Containerization seems poised to offer both a complement and a viable alternative to server virtualization**”

(1) IDC



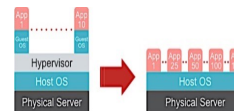
Containers vs VMs vs Bare-metal Servers

	Container	Virtual Machine	Bare-Metal x86 Server
Underlying Platform	OS on Virtual Machine or Bare-Metal x86 Server	Hypervisor on Bare-Metal x86 Server	N/A
Performance: Speed and Consistency	Average	Average	Fastest
Provisioning Time	Seconds	Minutes	Hours
Tenant Isolation Enforcement	OS Kernel	Hypervisor	Physical
Ideal Application Types	Mode 2	Mode 1 or Mode 2	Mode 1 or Mode 2
Configuration and Reconfiguration Flexibility	Highest	Medium	Lowest
Host Consolidation Density	Maximum	Average	None
Application Portability	Application Packaging/ Manifest*	VM Image, VM Migration Tools	Backup and Restore, ISO Images
Granularity	Extremely Small	Average	Largest
*While application portability is somewhat easier in container environments that are leveraging a container management and orchestration solution, portability should not assumed to be universal — differences in the underlying host OS below the containers could still present some interoperability challenges.			

Source: Gartner (September 2015)

Driving Factors for Containers

Density & Performance



Licensing Costs



Shift to DevOps



Cloud-native Applications
(Scale-out)

NETFLIX

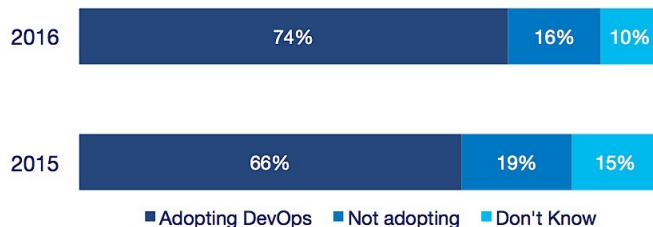
Faster Exploration & Deployment
(CI/CD)



“Containerization seems poised to offer both a **complement** and a **viable alternative** to server virtualization” - IDC

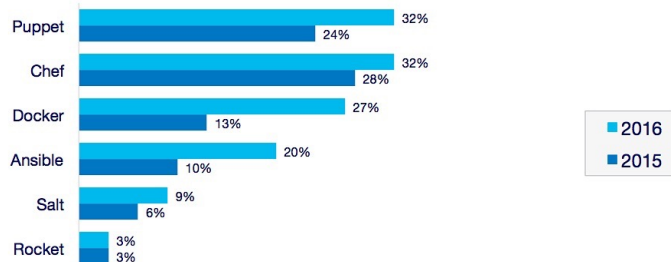
From Waterfall to DevOps

DevOps Adoption Up in 2016



Source: RightScale 2016 State of the Cloud Report

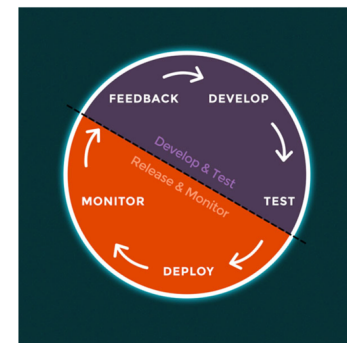
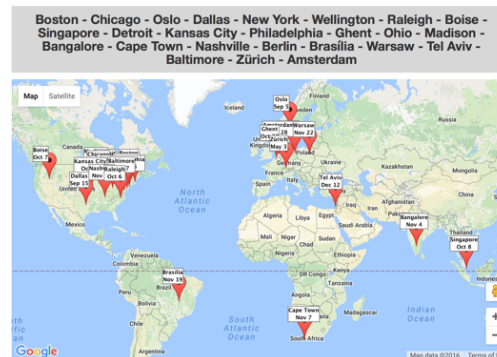
Respondents Using DevOps Tools



Source: RightScale 2016 State of the Cloud Report

“ In 2018 more than 50% of new Workloads will be deployed into containers in at least one stage of the life application cycle⁽¹⁾ ”

(1) Gartner

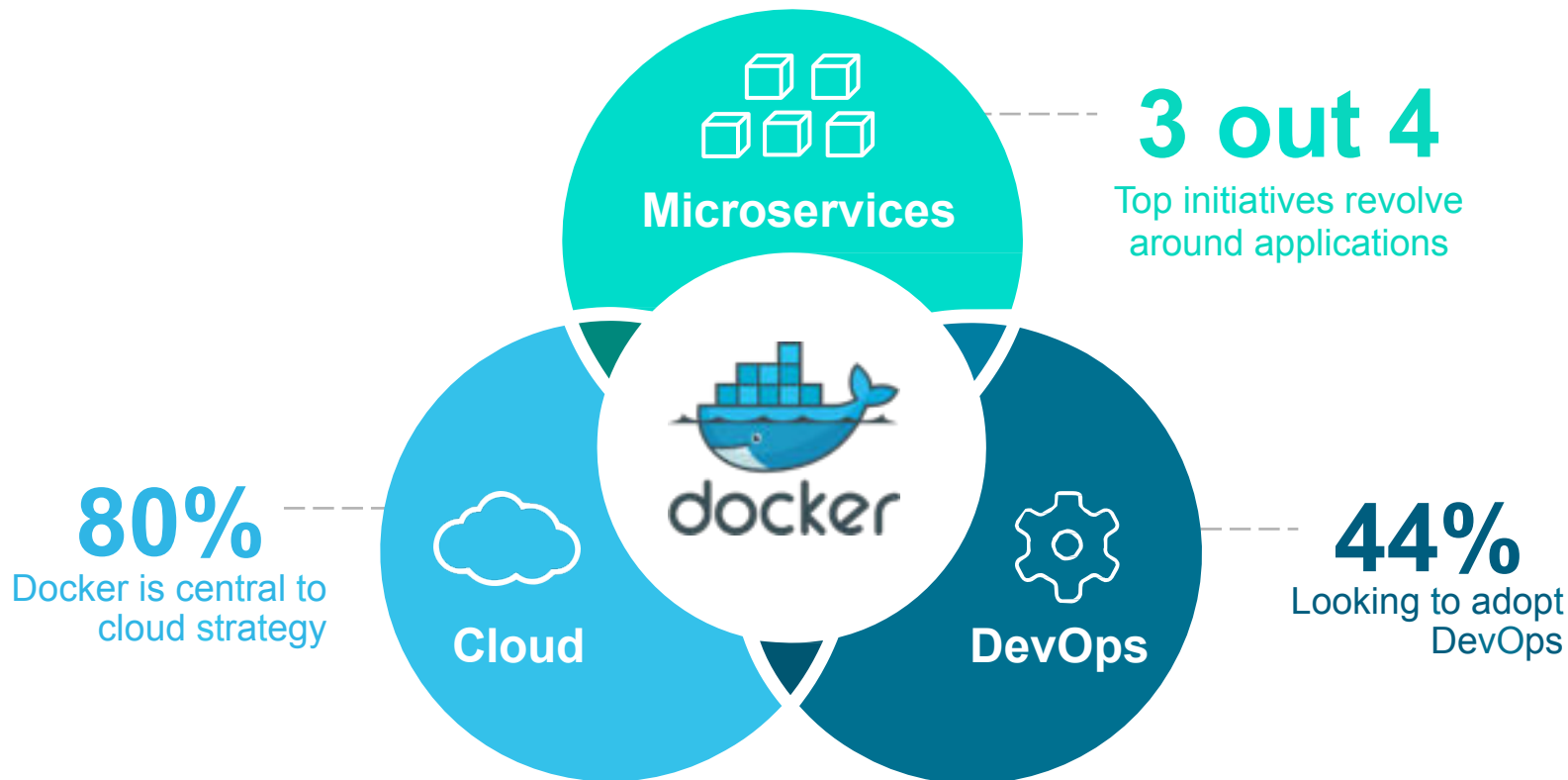


Poll Question #2

Does Your Organization Practice DevOps or Have a DevOps Department?

- A. Practicing DevOps
- B. Planning on Practicing DevOps
- C. No DevOps Plans

Driving force behind modern app initiatives

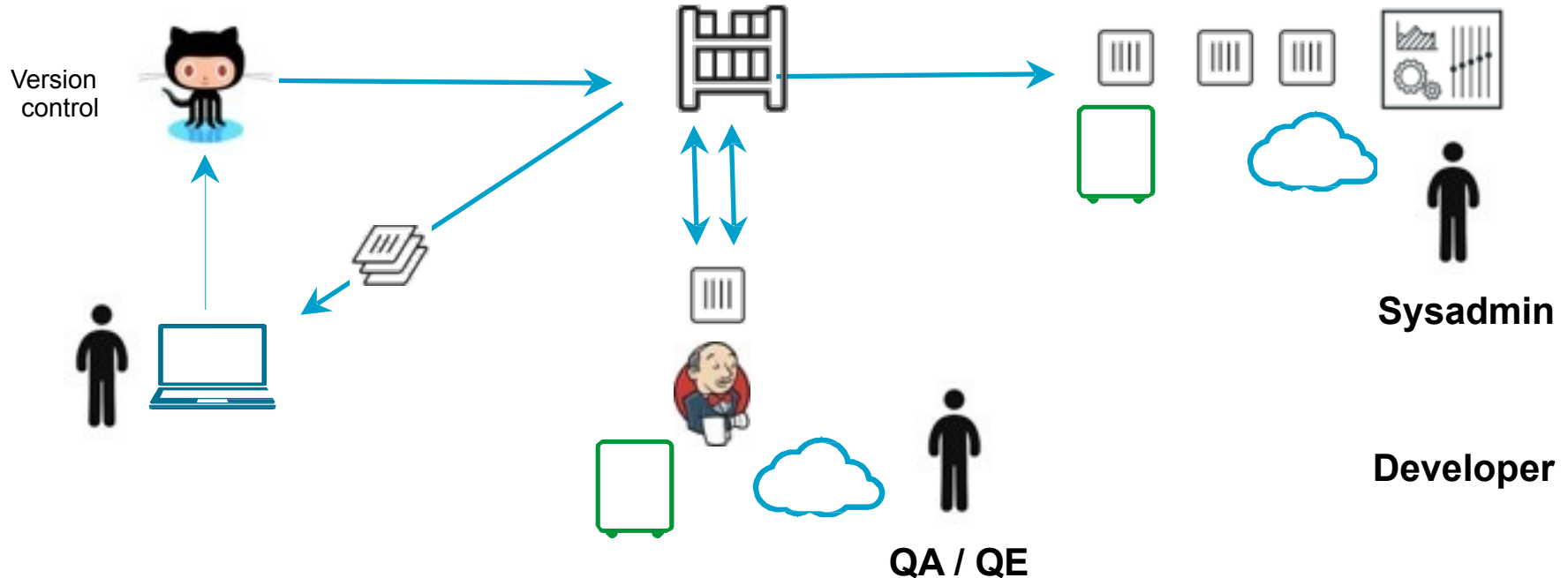


Scenario: Continuous Integration and Delivery

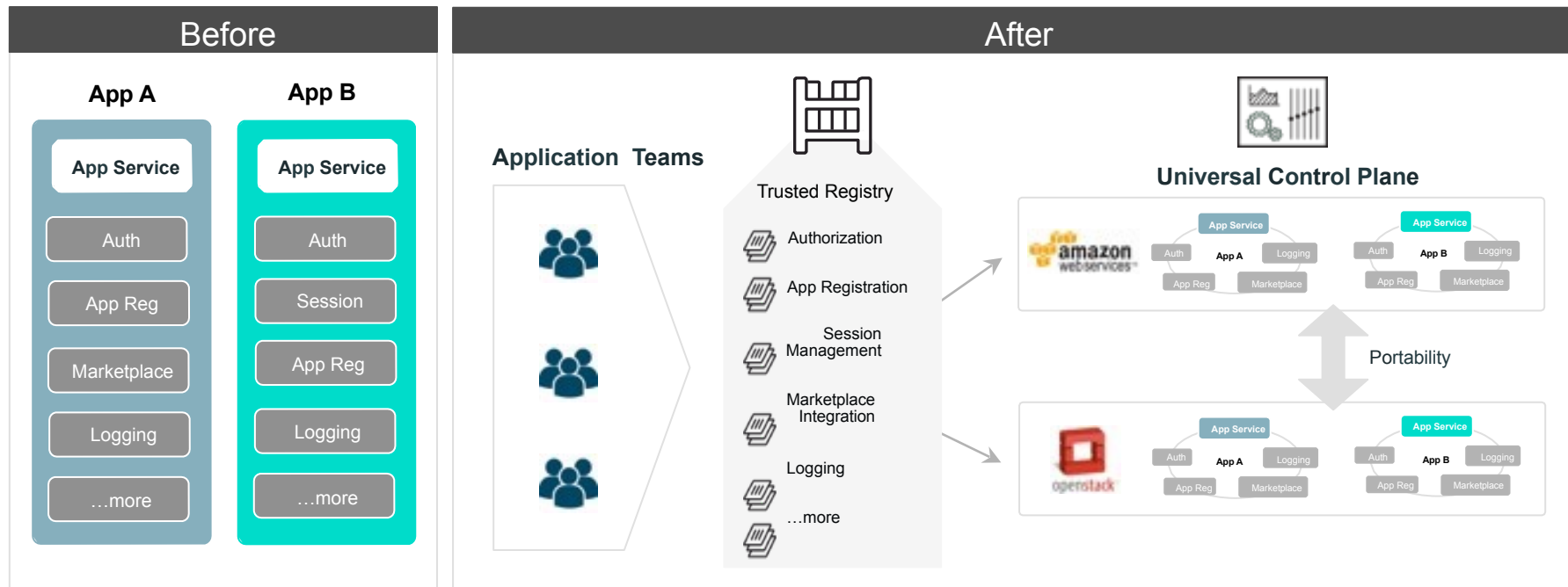
1. Development

2. Test

3. Stage / Production



Scenario: Enabling Transformation to Microservices



Common services in monoliths are turned into base applications stored in the Trusted Registry available to all app teams

Teams request into central IT maintained portal/registry to provision infrastructure and pull base images

Monoliths are now micro services applications. Each app has its own containers based on the same base image

What is Docker?

Poll Question #3

Are You Familiar with Docker?

- A. Yes
- B. No

What is Docker?

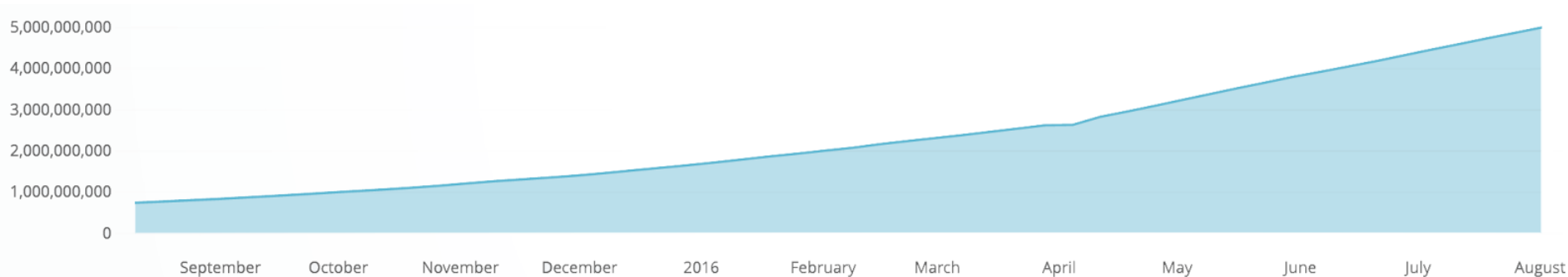
Docker is a platform for developing, shipping and running applications using container technology

The Docker Platform consists of multiple products/tools

- Docker Engine
- Docker Hub
- Docker Trusted Registry
- Docker Machine
- Docker Compose
- Docker for Windows/Mac
- Docker Datacenter

Docker Adoption

- Docker Hub - “Pull” is one download of a container image
- 650,000 registered users
- 5 Billion pulls since 2013
- Growing by 150% per month



Persistent Storage for Containers

Stateful vs Stateless

“**Stateful container apps represent the next big IT challenge⁽¹⁾**”

“**Persistent storage among top issues for container enterprise-readiness in production⁽²⁾**”

“**Stateful Database applications such as Redis, MySQL, MongoDB among most pulled images on Docker Hub⁽²⁾**”

- (1) Container Journal
(2) Gartner

Importance of Container Orchestration Abilities

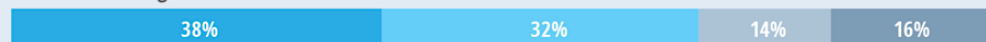
Long-running applications



Load balancing



Persistent storage



Batch applications



Non-containerized workloads



Extremely Important Very Important Moderately Important Slightly Important Not At All Important Not Applicable

The New Stack Survey, March 2016. When evaluating container orchestration tools, how important following requirements? n=107. Due to rounding, figures may not equal 100%.

Persistent Storage—Why

Data Accessibility

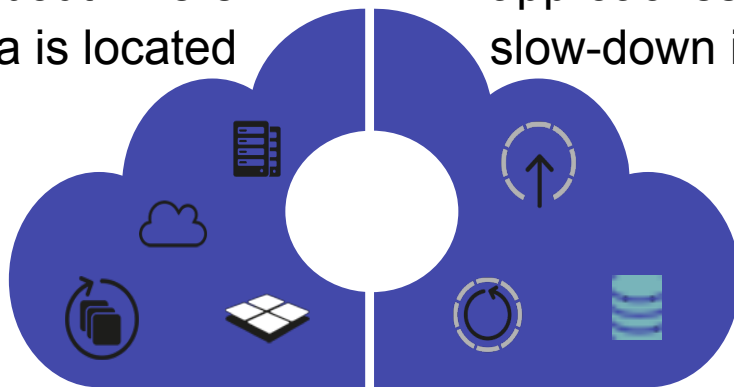
Run containers anywhere
without worries about where
data is located

Deployment Cycles

Traditional storage
approaches
slow-down innovation

Data Availability

Data needs to be
always on no matter
what happens



Storage Costs

Storage defined by
Software using any
commodity HW or Cloud

Storage Services for Containers

Persistent Storage
Management



Scale-Out



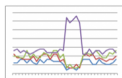
Snapshots



I/O Acceleration



Quality of Service



Encryption



Disaster Recovery



Heterogeneous
Support



Poll Question #4

Do you have storage challenges in running containerized applications in production?

- A. Persistent Storage
- B. High Availability
- C. Quality-of-Service
- D. All of these

Docker Storage Types

Registry

Cold storage of container images

Graph

Active storage of running container images

Volume

Persistent block storage for data

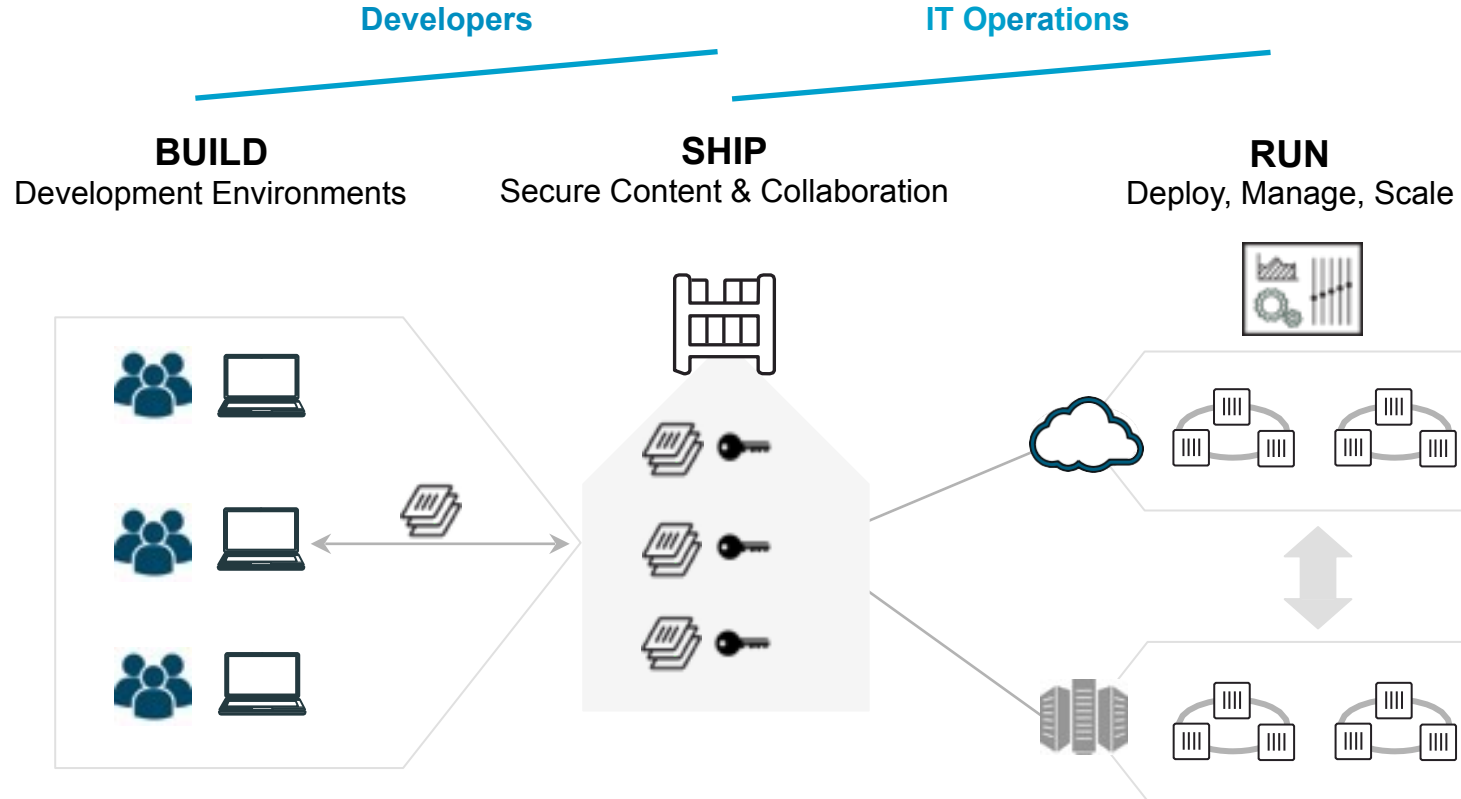
- Config data stored via standard Docker volumes
- Images stored via driver
- Native filesystem (We don't care what's beneath - NFS or iSCSI is best!)
- Drivers available for cloud object storage for images (S3, Swift, GCS)

Docker Volume Storage

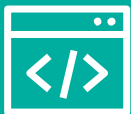
- This is where persistent data lives
- Extremely pluggable
- Network attached storage is extremely useful here

Docker Data Center (DDC)

Docker Datacenter CaaS workflow



DDC Delivers Agility, Portability and Control



Developers

- Freedom to create and deploy apps fast
- Define app needs



IT Operations

- Quickly and flexibly respond to changing needs
- Standardize, secure, and manage

Frictionless portability across teams, environments, infrastructure

Docker Datacenter delivers agility



- **Easy to setup and use**
 - Fastest time to value for ops
 - Developer self service from library of images
- **Native Docker solution**
 - Full support for Docker API
 - Integrated Docker Engine, Swarm, Compose
- **Extend the existing Docker developer experience to deployment**
 - Deploy Compose apps directly in UCP

Docker Datacenter delivers portability



Portability

- **Compute, network and storage**
 - Network and volume plugins ensure apps work without recoding new environments
 - Move across public, private and hybrid cloudility
- **Seamless dev to prod workflow**
 - Same code in dev runs in prod w/o changes
 - Eliminate the “works on my machine” issues
 - Full support for Docker API

Docker Datacenter delivers control



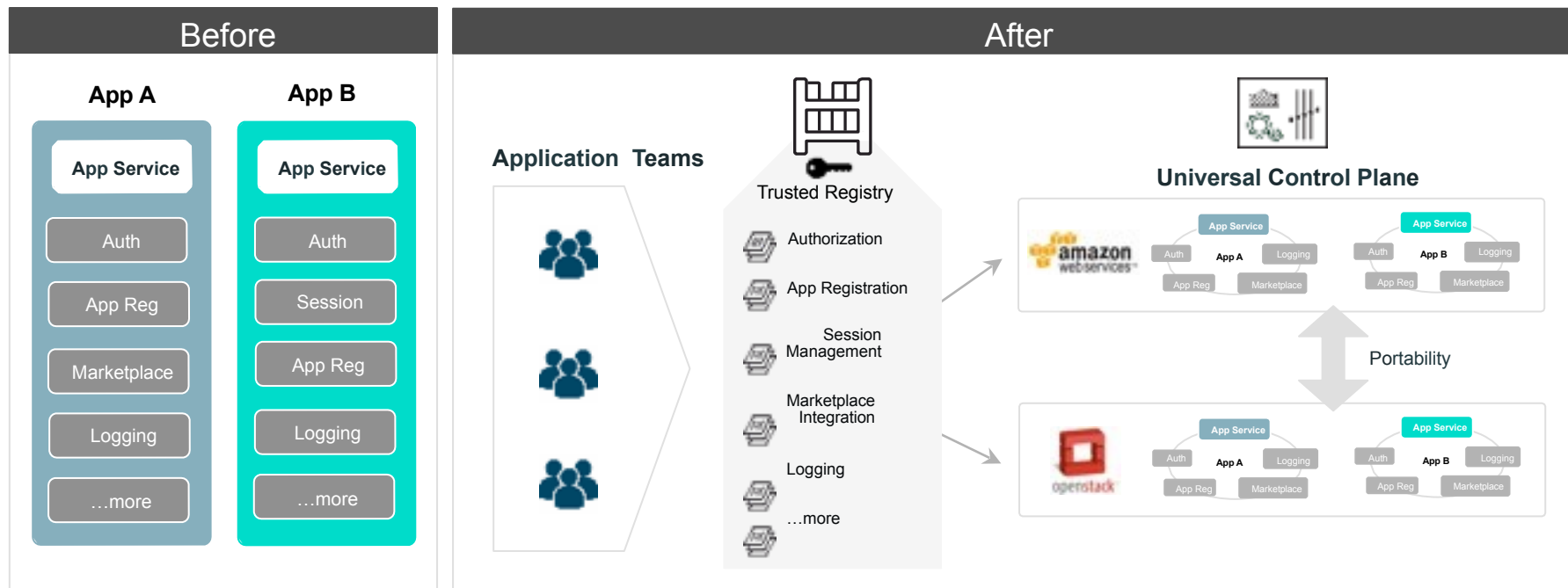
Control

- **Management ease at scale**
 - View and manage apps, containers, nodes, volumes, networks, images, users and groups
 - Built in HA, backup/restore
 - Point and click GUI or CLI support
- **Integrated security and enterprise controls**
 - Content Trust for image signing and verification
 - Secure access with RBAC and LDAP/ADP
 - Out of the box TLS
- **Extend and integrate DDC to your systems and processes**

Docker Case Studies:

Brief overviews of recent success stories for Docker customers

Case Study: ADP



Common services in monoliths are turned turned into base applications stored in the Trusted Registry available to all app teams

Teams request into central IT maintained portal/registry to provision infrastructure and pull base images

Monoliths are now micro services applications. Each app has it's own containers based on the same base image

Case Study: SA Home Loans



SA Home Loans uses Docker Datacenter to convert Monoliths to Microservices

Goal

- Convert monolithic .Net applications (built in Mono) into microservices

Result

- Evaluated Docker running small-scale postgres services across 2 nodes
- Docker's enterprise-class networking and security capabilities were key but impressed with the ease-of-use of Docker Native orchestration - Swarm
- Currently running Docker Datacenter across 4 nodes as they are working to Dockerize all enterprise-class applications in the next few months

Case Study: GSA (Booz Allen)



Challenge

- Migrate away from monolithic application
- Long and cumbersome application development cycles

Solution

- Build a new developer platform (IAE Common Service Platform) with Docker Trusted Registry and commercially supported Docker Engine on AWS

Benefit

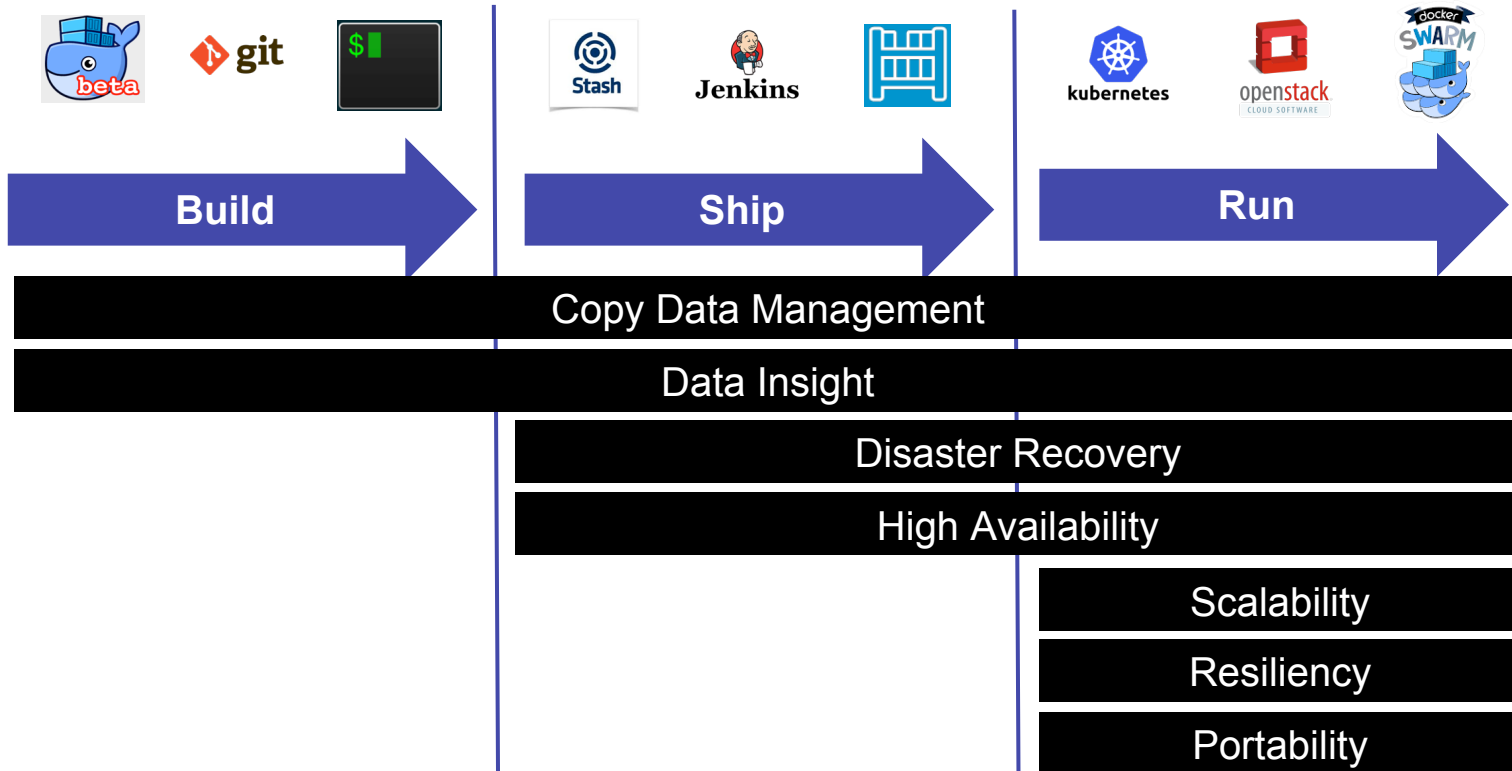
- Improved customer centric services Reduced time-to-market
- Improve security and reduce attack surface area



Container Future Areas of Exploration

Build | Ship | Run

Information Created



But Wait, There's More!

▶ Please join us for part-two of this Webcast!

“Containers: Best Practices & Data Management Services”

December 7, 2016, 10:00 am PT

▶ Register at:

<https://www.brighttalk.com/webcast/663/227349>

▶ Stay updated! Join our Containers opt-in email List

<http://eepurl.com/ciMk0P>

- The SNIA Ethernet Storage Forum (ESF) is hosting a Containers Webcast next month

“The State of Storage in the Container World”

November 17, 2016, 10:00 am PT

- Register at:
<https://www.brighttalk.com/webcast/663/225901>

After This Webcast

- Please rate this Webcast. We value your feedback
- This Webcast and a copy of the slides will be on the SNIA Cloud Storage website and available on-demand
 - ◆ <http://www.snia.org/forum/csi/knowledge/webcasts>
- A Q&A from this webcast, including answers to questions we couldn't get to today, will be on the SNIACloud blog
 - ◆ <http://www.sniacloud.com/>
- Follow us on Twitter @SNIACloud



Thank you.

Keith.Hudgins@docker.com

Chad.Thibodeau@veritas.com