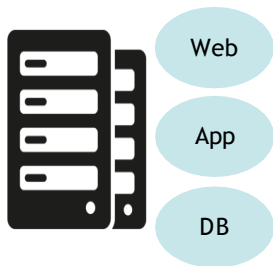




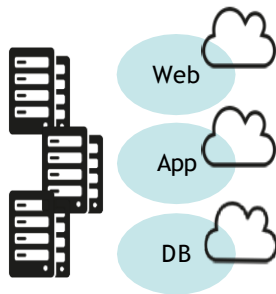
CONTAINERS AND DOCKER

APPLICATION DEPLOYMENT HISTORY

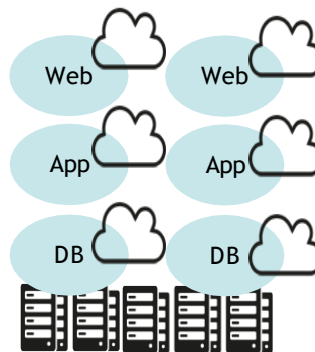
Monolithic Apps on Physical



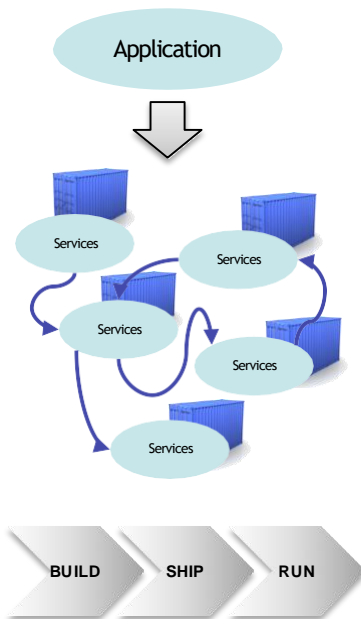
Virtual Machine Abstraction



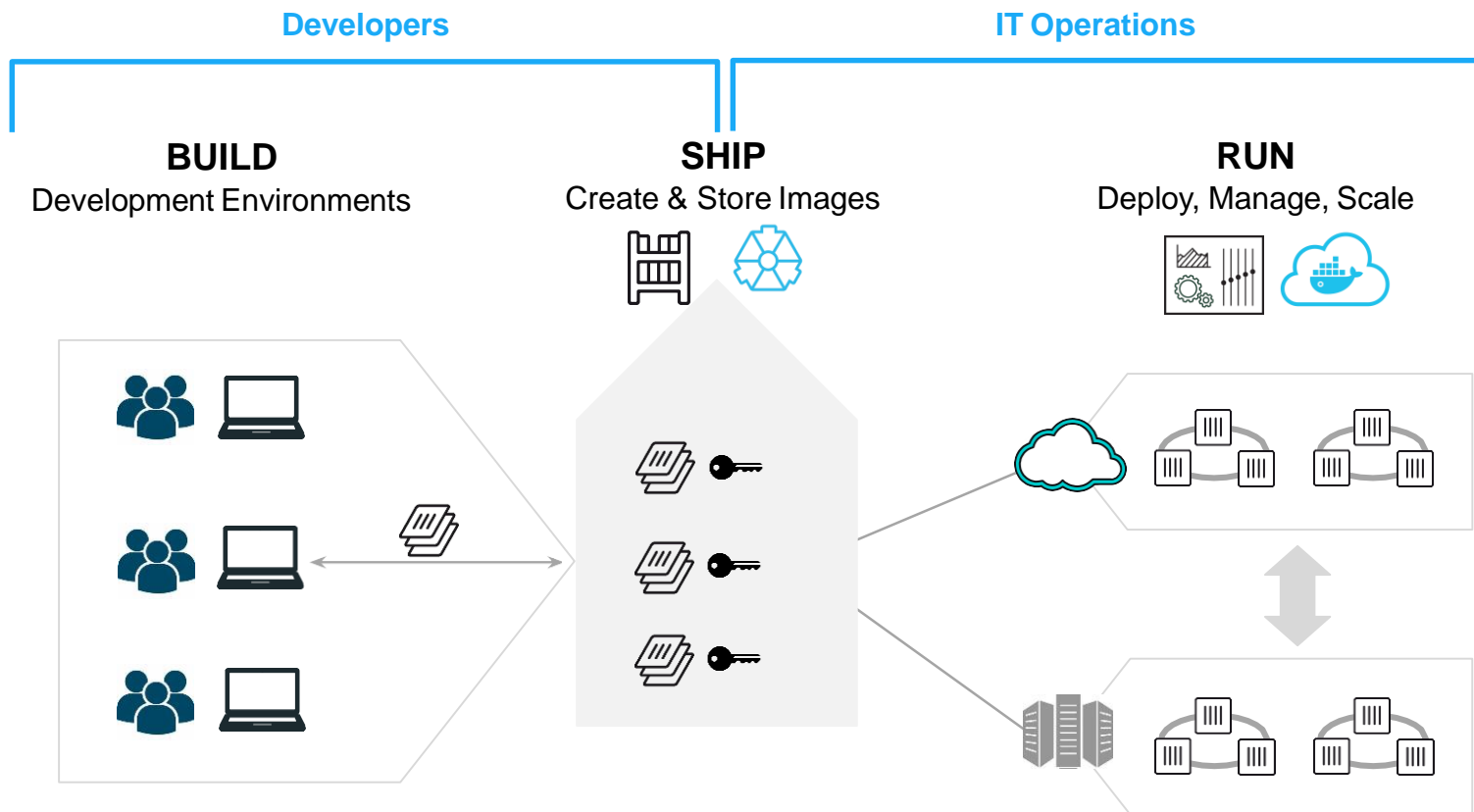
Stateless & Horizontal Scalable Apps



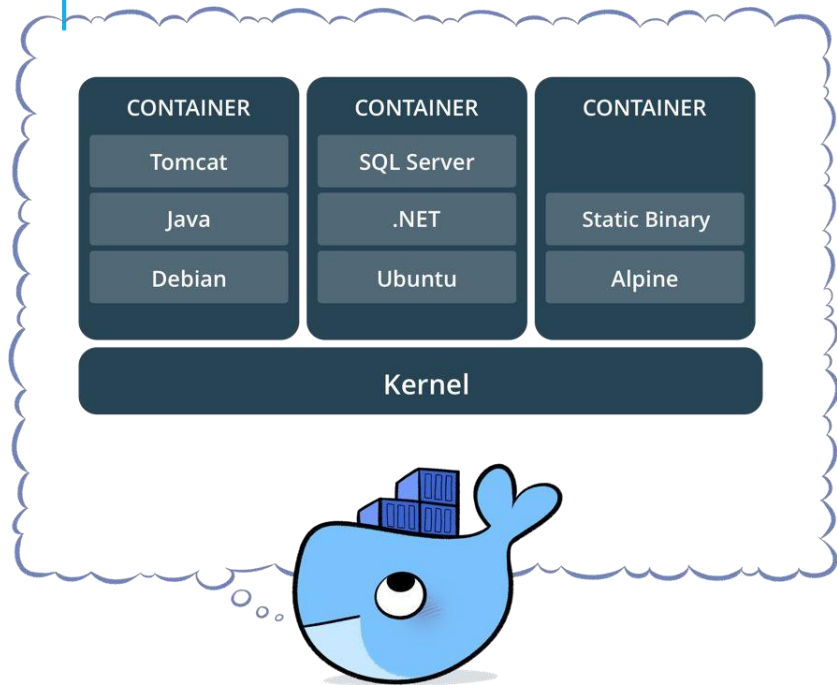
Micro-services & Containers



USING DOCKER: BUILD, SHIP, RUN WORKFLOW



WHAT IS A CONTAINER?



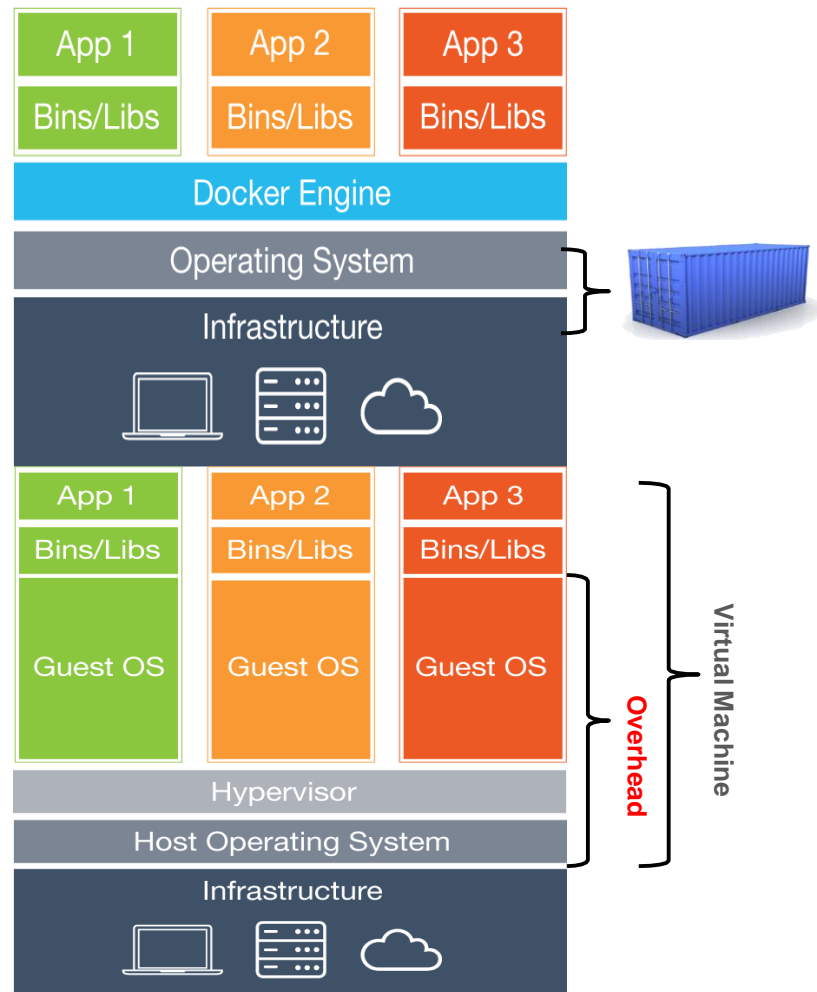
- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works for all major Linux distributions
- Containers are now native to many popular OS

CONTAINERS

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”



THE ROLE OF IMAGES AND CONTAINERS



Docker Image

Example: Ubuntu with Node.js and
Application Code



Docker Container

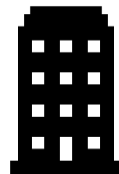
Created by using an image. Runs
your application.

DOCKER CONTAINERS ARE NOT VMS

- Easy connection to make
- Fundamentally different architectures
- Fundamentally different benefits



Virtual Machine



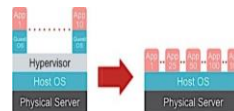
Containers

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 be assumed to be universal — differences in the underlying host OS below the containers could still present some interoperability challenges.			

DRIVING FACTORS FOR CONTAINERS

Density & Performance



Licensing Costs



Shift to DevOps



Cloud-native Applications
(Scale-out)

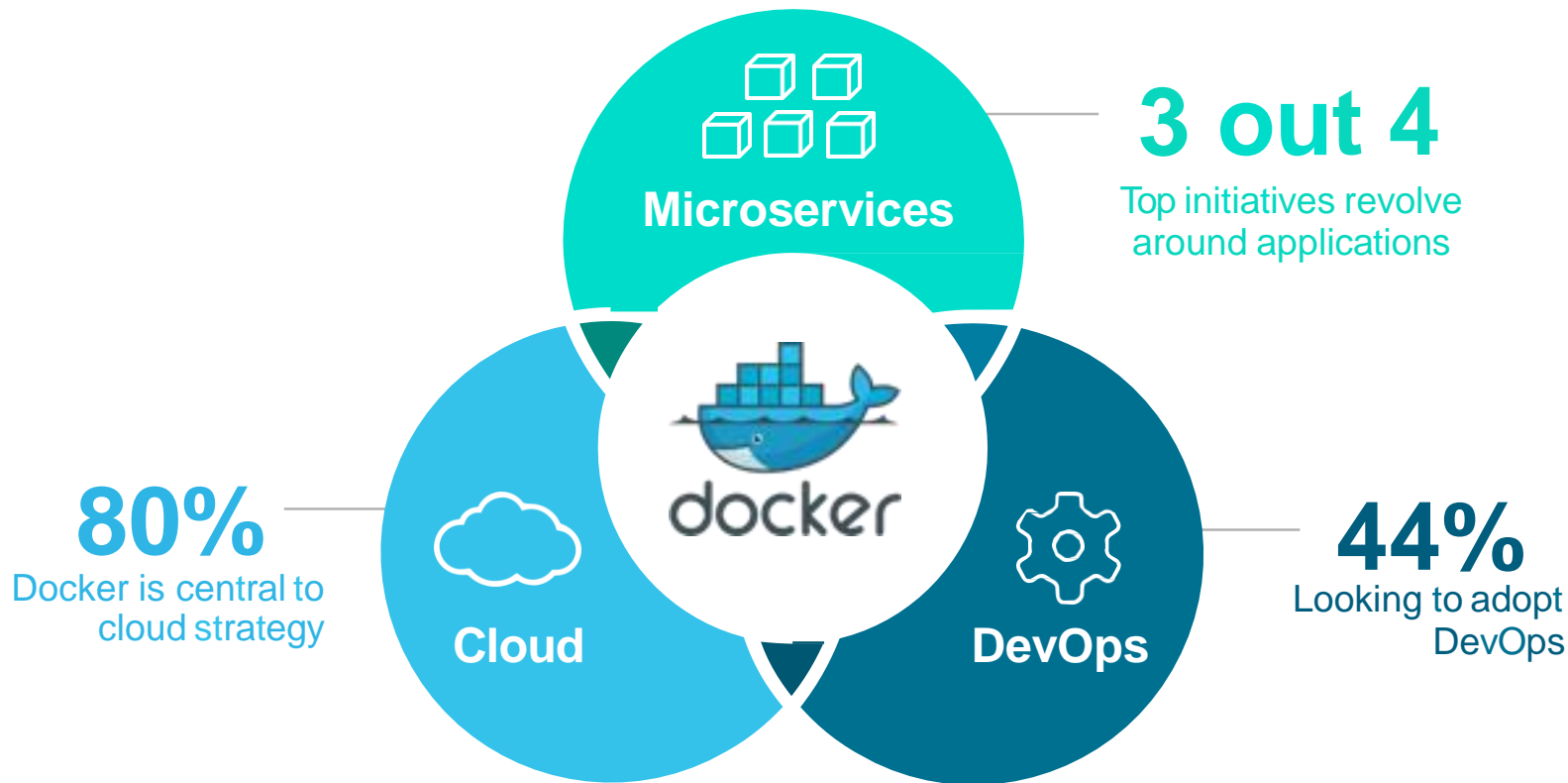
NETFLIX

Faster Exploration & Deployment
(CI/CD)



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

DRIVING FORCE BEHIND MODERN APP INITIATIVES

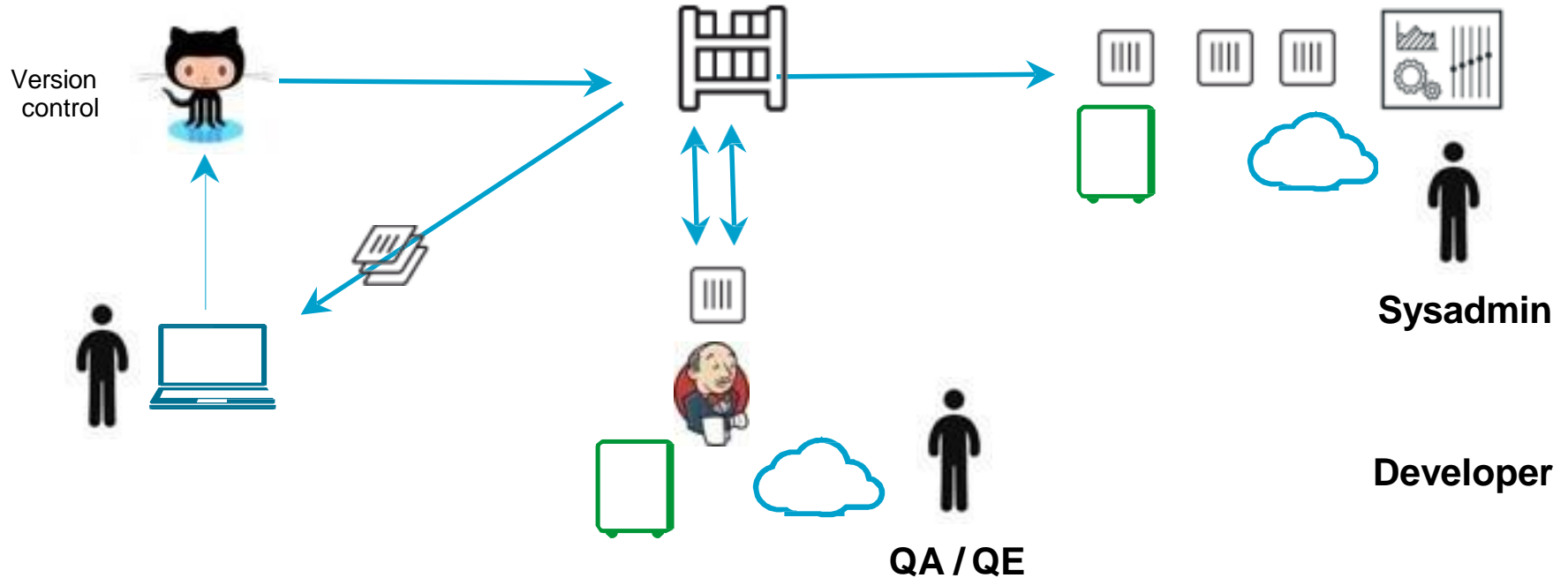


SCENARIO: CONTINUOUS INTEGRATION AND DELIVERY

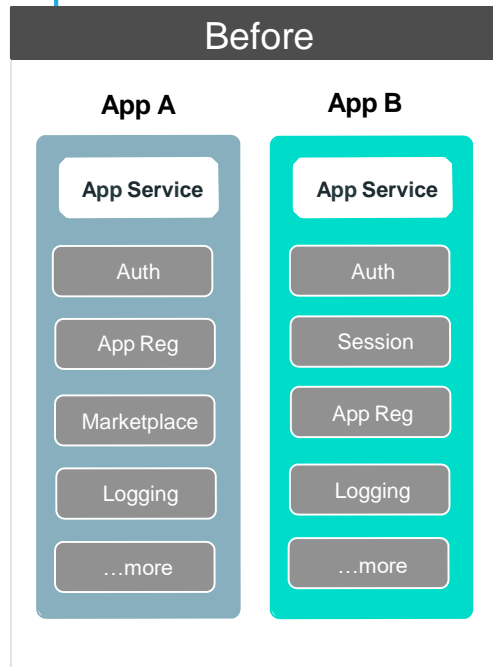
1. Development

2. Test

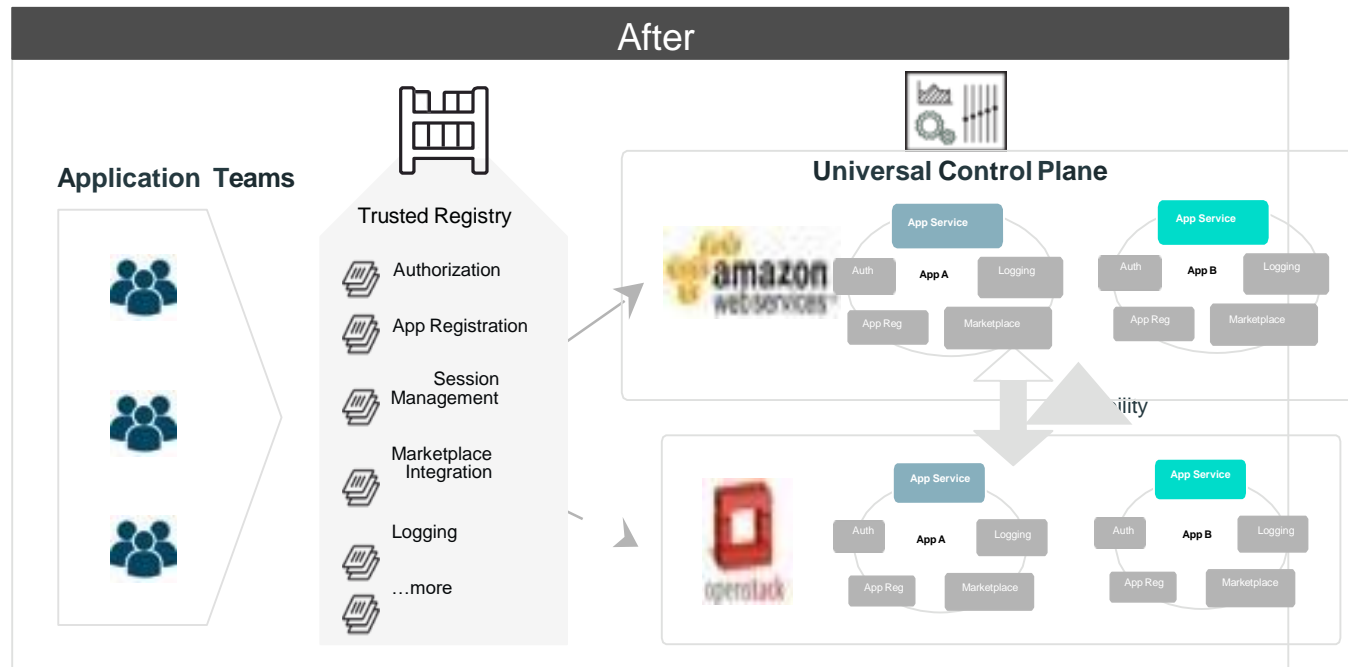
3. Stage / Production



SCENARIO: ENABLING TRANSFORMATION TO MICROSERVICES



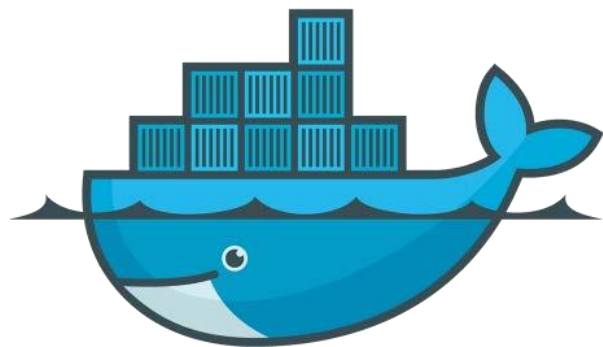
Common services in monoliths are turned into base applications stored in the Trusted Registry available to all app teams



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?



- LIGHTWEIGHT, OPEN, SECURE PLATFORM
- SIMPLIFY BUILDING, SHIPPING, RUNNING APPS
- Runs natively on Linux or Windows Server
- Runs on Windows or Mac Development machines (with a virtual machine)
- Relies on "images" and "containers"

WHAT IS DOCKER?

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

The Docker Platform consists of multiple products/tools

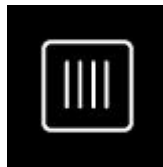
- Docker Engine
- Docker Hub
- Docker Trusted Registry
- Docker Machine
- Docker Compose
- Docker for Windows/Mac
- Docker Datacenter

SOME DOCKER VOCABULARY



Docker Image

The basis of a Docker container. Represents a full application



Docker Container

The standard unit in which the application service resides and executes



Docker Engine

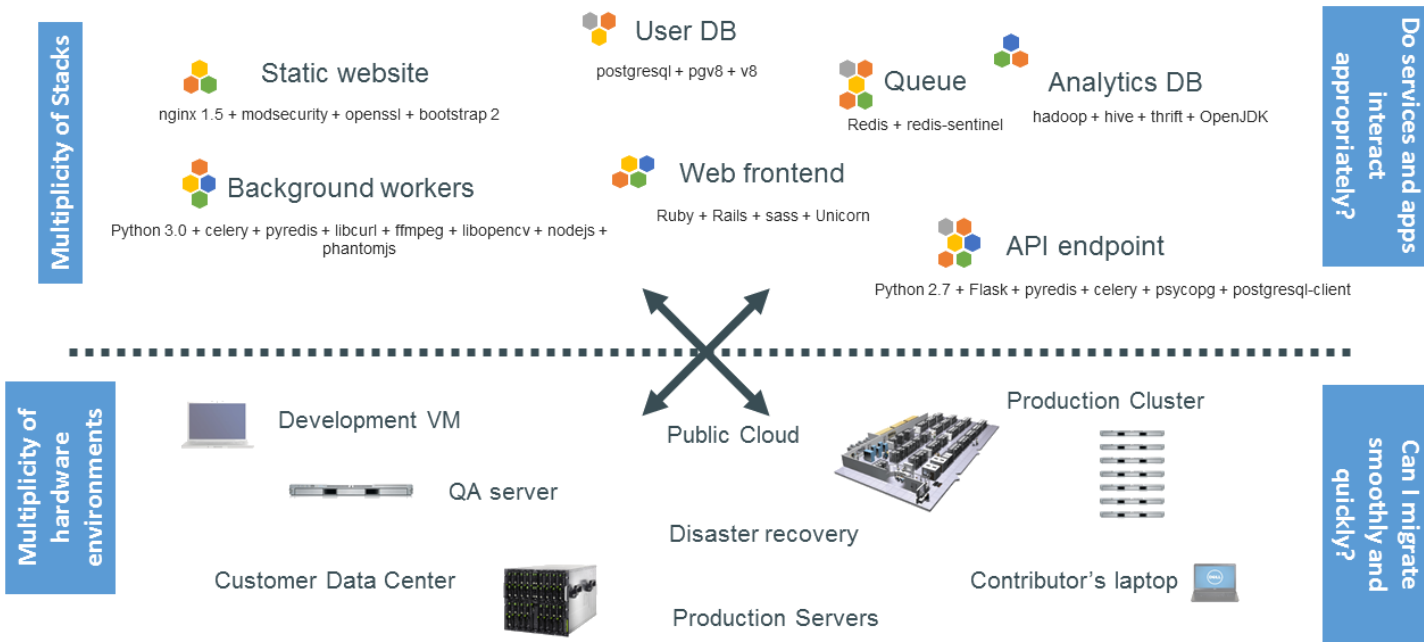
Creates, ships and runs Docker containers deployable on a physical or virtual, host locally, in a datacenter or cloud service provider




Registry Service (Docker Hub(Public) or Docker Trusted Registry(Private))










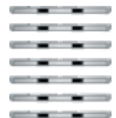



Cloud or server based storage and distribution service for your images

THE CHALLENGE.....



THE MATRIX FROM



	Static website	?	?	?	?	?	?	?
	Web frontend	?	?	?	?	?	?	?
	Background workers	?	?	?	?	?	?	?
	User DB	?	?	?	?	?	?	?
	Analytics DB	?	?	?	?	?	?	?
	Queue	?	?	?	?	?	?	?
		Development VM	QA Server	Single Prod Server	Onsite Cluster	Public Cloud	Contributor's laptop	Customer Servers
								

CARGO TRANSPORT PRE-1960.....

Multiplicity of Goods
















Do I worry about
how goods interact
(e.g. coffee beans
next to spices)

Multiplicity of
methods for
transporting/storing



Can I transport quickly
and smoothly
(e.g. from boat to train
to truck)

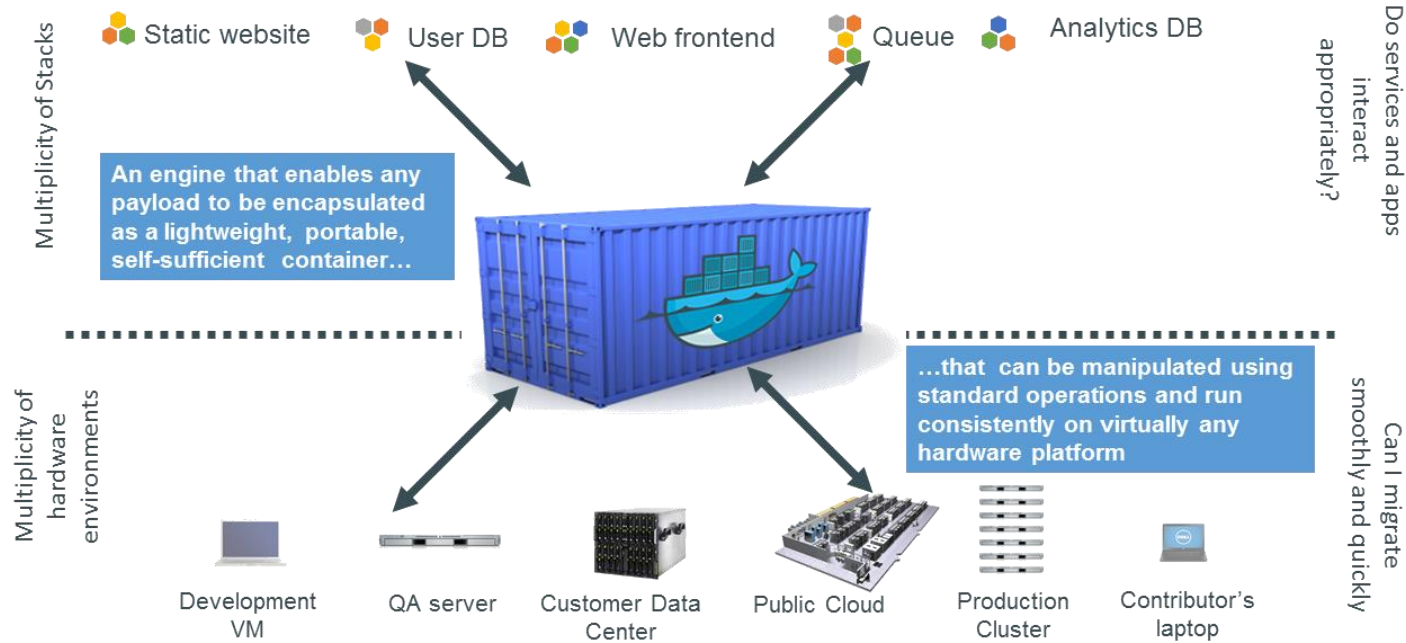
ALSO A MATRIX FROM

	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
	?	?	?	?	?	?	?
							

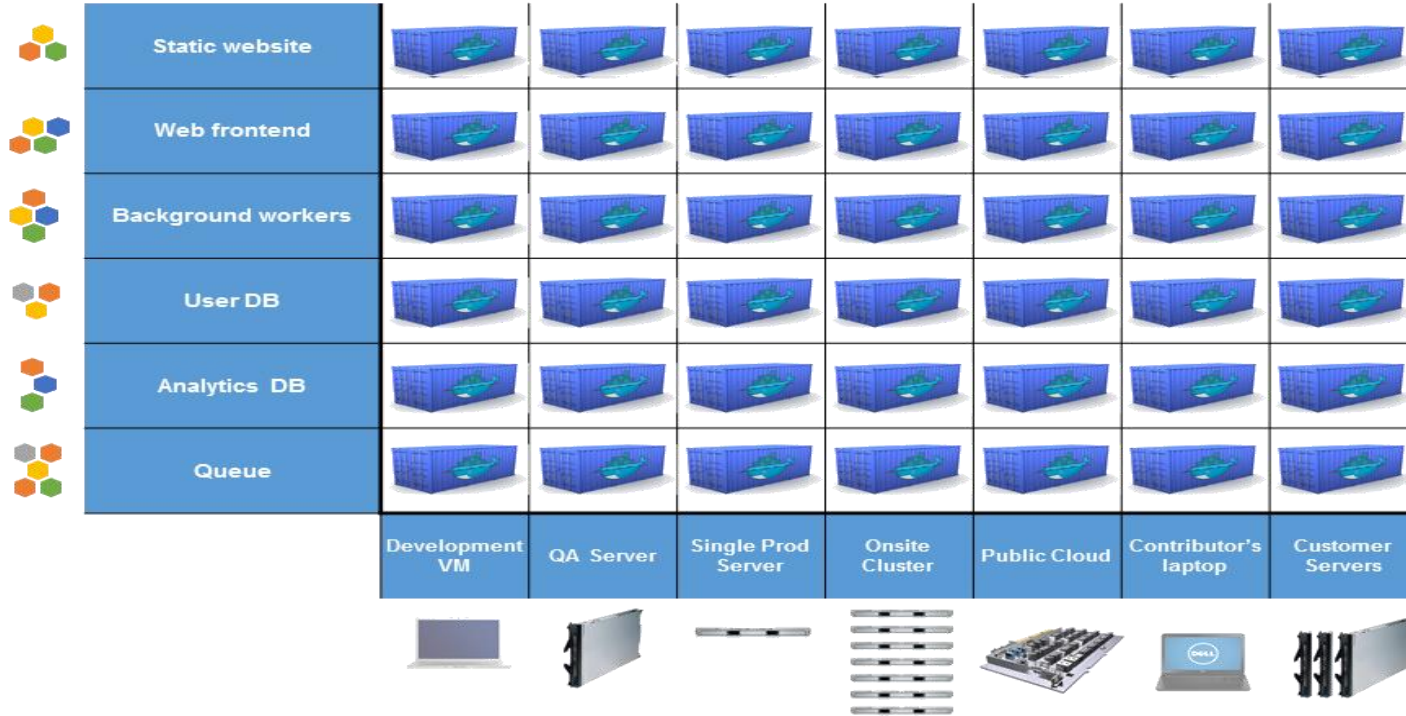
SOLUTION: INTERMODAL SHIPPING CONTAINER.....



DOCKER IS A CONTAINER SYSTEM FOR CODE.....



DOCKER ELIMINATES THE MATRIX

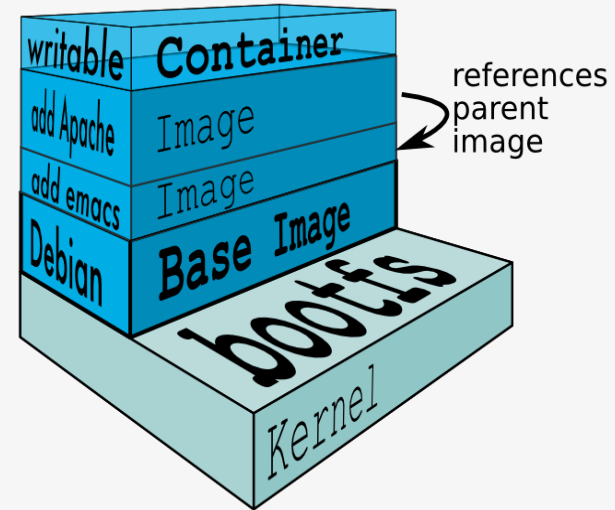


Docker Technology

libvirt: Platform Virtualization

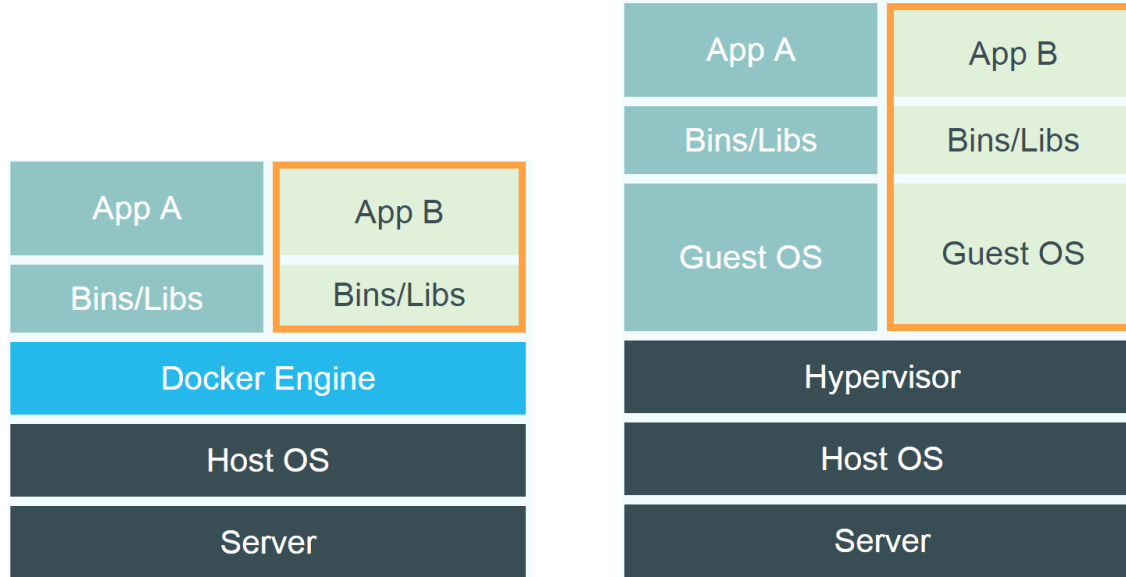
LXC (Linux Containers): Multiple isolated Linux systems (containers) on a single host

Layered File System



[Source: <https://docs.docker.com/terms/layer/>]

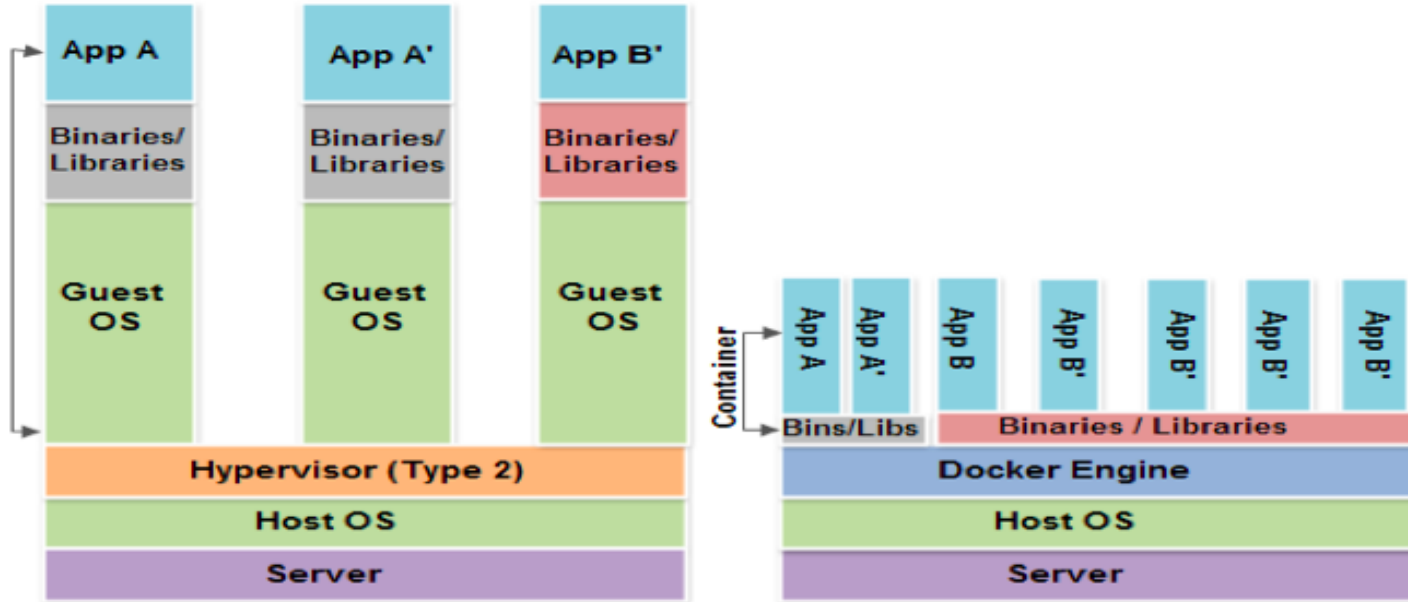
Docker vs. Virtual Machine



Source: <https://www.docker.com/whatisdocker/>

VIRTUAL MACHINE VERSUS CONTAINER.....

Containers vs Virtual Machines

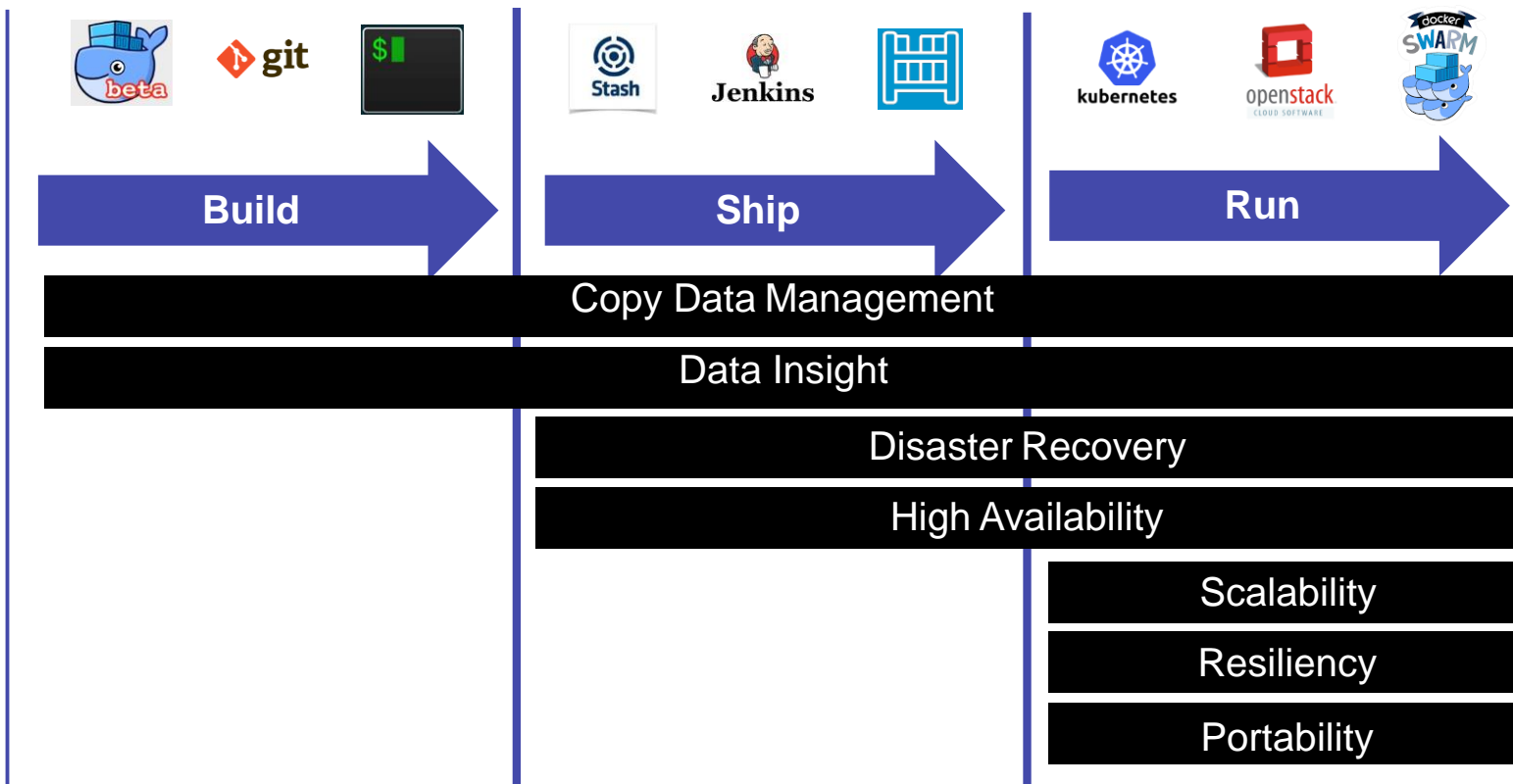




CONTAINER AREAS OF EXPLORATION

BUILD | SHIP | RUN

Information Created



Exercise

