CONTAINERS IN ACTION
Transform Application Delivery with DevOps

Daniel Oh
AppDev Solution Architect
Agile & DevOps CoP Manager
ABOUT ME

- AppDev Solution Architect, Agile & DevOps CoP Manager at Red Hat
- Technology Consultant at Accenture
- Application & Software Architect at Samsung

- JBoss Middleware
- OpenShift Container Platform
- Mobile Application Platform
AGENDA

- Problem
- The Solution
- DevOps with the Solution
- Practical Guidelines
I.T. delivers thousands of applications to meet the needs of the business.

These applications require complicated collaboration during installation and integration every time they are deployed.
Many have different requirements.

They also use different languages, databases, and tools.
THE PROBLEM

To deploy, configure, manage, and maintain this complexity takes:

- People,
- expertise,
- and the right systems, infrastructure, and architecture.

This costs time and money.
THE PROBLEM

Applications require complicated installation and integration every time they are deployed leading to

- Slow service delivery
- Reduced service quality
- Frequent down times
THE PROBLEM

DEVELOPERS

I.T. OPERATIONS
THE SOLUTION

Adopting a container strategy will allow applications to be easily shared and deployed

- Consistent env and tools
- Predictable building blocks
- Faster deployment
Containers have arrived in the Enterprise

- 48% of companies are using containers today, up 30% from 1 year ago
- 5B downloads according to Docker
- #1 requested topic at Red Hat Summit

Containers are Linux technology and we have a massive install base with RHEL
Containers - Transform Apps, Infrastructure & Process

Application Architecture
- Monolithic
- N-Tier
- Microservices

Development Process
- Waterfall
- Agile
- DevOps

Application Infrastructure
- Datacenter
- Hosted
- Cloud

Containers

Transform Apps, Infrastructure & Process through Containers.
WHAT ARE CONTAINERS?
It Depends Who You Ask

<table>
<thead>
<tr>
<th>INFRASTRUCTURE</th>
<th>APPLICATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Linux operating system features</td>
<td>● A self-contained package</td>
</tr>
<tr>
<td>● Have Linux inside</td>
<td>● Easily add to already built applications</td>
</tr>
<tr>
<td>● Simpler and lighter than virtual machines</td>
<td>● Easily share containerized components</td>
</tr>
<tr>
<td>● High density</td>
<td>● Everything an app needs to run is inside</td>
</tr>
<tr>
<td>● Need orchestration and management</td>
<td>● More control for developers</td>
</tr>
</tbody>
</table>
THE SOLUTION

Hardware

Virtual Machine

Operating System

Container

App

Controlled by
Developers

Controlled by
IT Operations
THE SOLUTION

DEVELOPERS

I.T. OPERATIONS
THE SOLUTION

$ docker build -t app:v1 .
$ docker build -t app:v1 .

$ docker run app:v1
DEVOPS WITH CONTAINERS

dev → source repository → CI/CD engine → container → target environment
DEVOPS WITH CONTAINERS

dev → source repository → CI/CD engine → container

physical
virtual
private cloud
public cloud
DEVOPS WITH CONTAINERS
DEVOPS WITH CONTAINERS

$ docker build -t app/frontend:v1 .
$ docker build -t app/backend:v1 .
$ docker build -t app/database:v1 .
$ docker build -t app/cache:v1 .
$ docker build -t app/messaging:v1 .
DEVOPS WITH CONTAINERS

$ docker run app/frontend:v1 link-to-backend
$ docker run app/frontend:v1 link-to-backend
$ docker run app/backend:v1 link-to-db-cache-messaging
$ docker run app/backend:v1 link-to-db-cache-messaging
$ docker run app/database:v1
$ docker run app/cache:v1 link-to-db
$ docker run app/messaging:v1
DEVOPS WITH CONTAINERS
WE NEED MORE THAN JUST CONTAINERS

- **Scheduling**: Where should my containers run?
- **Lifecycle and health**: Keep my containers running despite failures
- **Discovery**: Where are my containers now?
- **Monitoring**: What’s happening with my containers?
- **Auth{n,z}**: Control who can do things to my containers
- **Aggregates**: Compose sets of containers into jobs
- **Scaling**: Making jobs bigger or smaller

WE NEED MORE THAN JUST CONTAINERS
CONTAINER ORCHESTRATION

Docker Swarm  kubernetes  Mesos
INDUSTRY CONVERGING ON KUBERNETES

Google Container Engine

OPENSHIFT®
by Red Hat®

MESOSPHERE

DEIS

apprenda

APCERA

TECTONIC
by CoreOS
DEVOPS WITH CONTAINERS AND ORCHESTRATION
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need networking
DEVOPS WITH
CONTAINERS AND ORCHESTRATION

Not enough! Need an image registry
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need metrics and logging
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need complex deployments e.g. A/B and Blue/Green
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need application lifecycle management
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need application services e.g. database and messaging
DEVOPS WITH CONTAINERS AND ORCHESTRATION

Not enough! Need self-service portal
Container application platform based on Docker and Kubernetes for building, distributing and running containers at scale
DEVOPS WITH OPENSSHIFT
DEVOPS WITH OPENSSHIFT

dev → source repository → CI/CD engine → container

Red Hat
OpenShift

Red Hat
Gluster
Storage

Red Hat
CloudForms
Focus on continuous integration first, then build out the pipeline
Start small: one application, one team
Pick the low-hanging fruits
Determine metrics and measure
Outcome-centric over output-centric metrics

PRACTICAL GUIDELINES

- Change lead time
- Change failure rate
- Mean time to repair
- Lines of code
- Number of features
- Code coverage
PRACTICAL GUIDELINES

- Work iteratively and deliver business value as quickly as possible
- Expand the deployment pipeline one step at a time
- Demonstrate measurable success, promote and celebrate it
- Evangelize to other teams and solicit volunteers for expansion
3 TAKEAWAYS FROM THIS SESSION

These are the things I want you to remember from my presentation

● Containers enable DevOps which leads to quality and speed

● Using containers across the entire delivery cycle has challenges that require more than just containers

● Red Hat OpenShift provides the fastest way to DevOps by providing a platform for building, deploying and managing containers at scale

TRY IT FOR FREE AT
www.openshift.com/devpreview
THANK YOU

+OpenShift
rhopenshift
facebook.com/openshift
@OpenShift