RED HAT SUMMIT

Delivering On-Premise Cloud With OpenShift

Jerry Eshbaugh
President - The Strategic Product

May 9th 3:30-> 4:15 PM
Moscone West Room 2003
Session Code: S1820

Why?

On-premise Cloud

- Why are we here?
- Why is this important?
- Why cloud technology in a traditional datacenter?
- Why is is this so hard?

Your Perspective

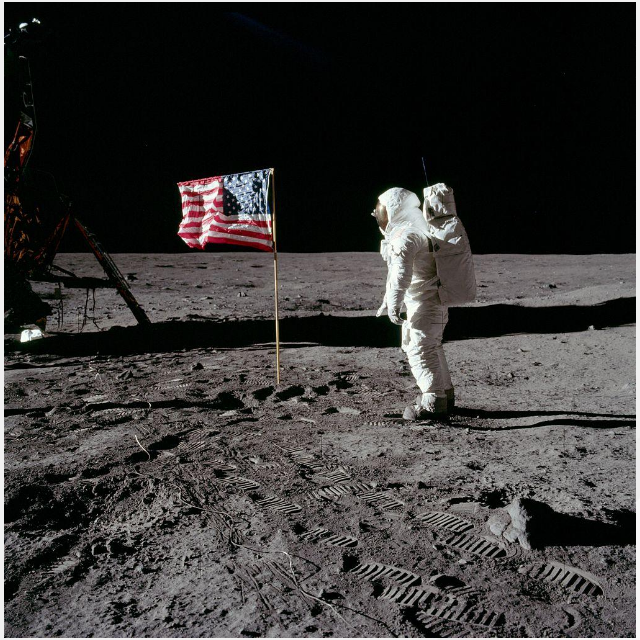
It has been great talking with some of you earlier
Here is what we have learned...

- A 100% cloud native initiative is not an option for all use cases
- Avoid cloud hosting cost escalation and vendor lock-in
- Support legacy applications that are not “cloud ready”
- Can I containerize mission critical applications and databases?

“Hybrid hosting is the reality for the majority of large organizations”

About Me

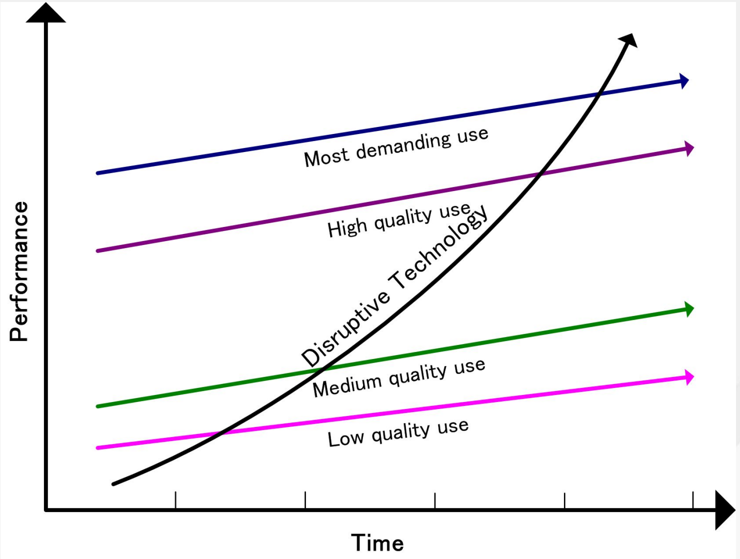
- Seeing the space mission as a child inspired me that anything is possible
- Frustration with the slow pace for innovation to deliver value
- On a mission to eliminate blockers
  - Cumbersome processes -> Automation
  - Poor quality -> Testing
  - Lack of visibility -> Monitoring
  - Barrier of empire -> OpenShift
- Experience
  - DevOps lead for USDA/DHS/FEMA
  - Serial entrepreneur
  - Strategic advisor



Why is this important?

We are in the middle of the biggest IT disruption since server virtualization

- DevOps
- Ubiquity of agile
- Cloud
- Test driven development, security and operations
- Continuous
- Bring your own device
- Big data
- Artificial intelligence
- Heightened end-user expectations



Why Cloud in Data Center?

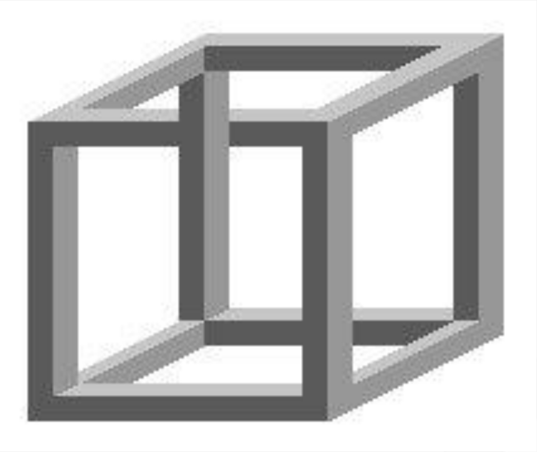
Disrupt or be Disrupted

- Deploy modern service based architectures on premise
- Integration with legacy applications
- Edge cases where security truly is an issue
- Predictable cost model
- Avoid hosting vendor lock-in
- Ownership is 9/10s
- Streamlined certification process

“Services like AWS are great if you have no legacy technology OR legacy culture”

Why is this so hard?

- Lack of visibility
- Fear of the unknown, change and disruption
- We have always done it that way
- Blame-o-saurus
- Silos
- Culture



“Things are rarely how they initially appear”

Culture Clash

Traditional/OnPrem

- Assembly line practitioner
- Hours worked / tickets closed
- Manually process
- Secure
- Rigid

DevOps/Cloud

- Multi-faceted engineer / architect
- Customer value delivered
- Automation
- Secure
- Agile

“Actions are driven reward and bound by fear”

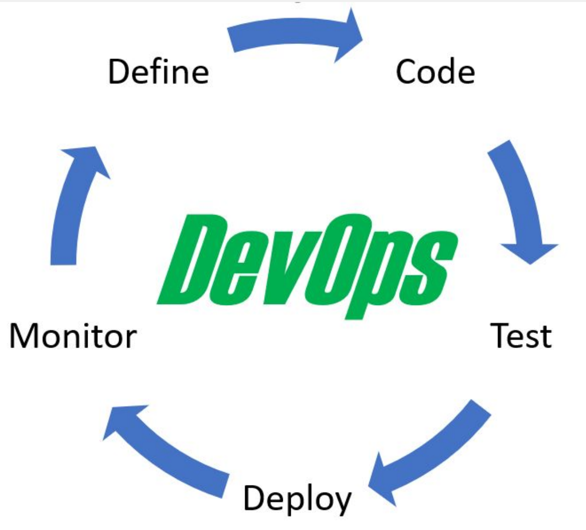
The Case for Automation

More Work

More Possibilities

Continuous

- Testing
- Integration
- Monitoring
- Delivery
- Improvement



“Frequency of deployment is the Net Promoter score of DevOps”

Case Study

Requirements

National Flood Insurance Modernization

- Agile processes
- Cloud technology
- Modernization effort

… Government owned and operated hosting facility

NATIONAL FLOOD
INSURANCE PROGRAM

On Premise Hosting Environment

Rigid

- VMware server farm
- RHEL 7.x
- NetApp Storage
- Physical and Logical network

OpenShift Cluster

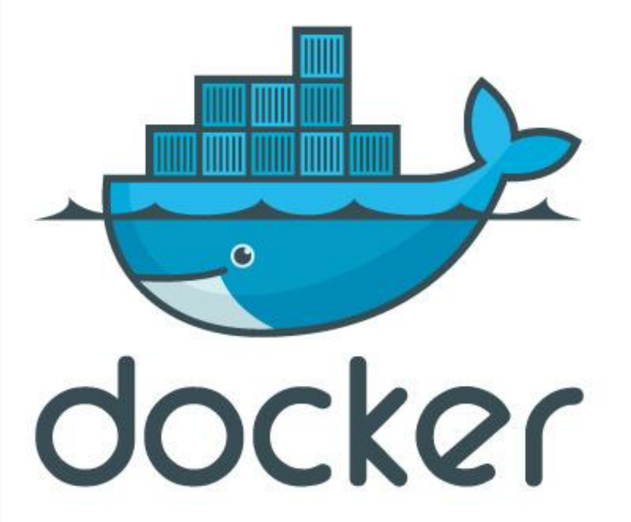
Agile

- OpenShift 3.x
- Ansible server configuration
- Projects (Environments)
  - CI-CD
  - Alpha
  - Smoke
  - Test
  - UAT
  - Stage
  - Validation
  - Prod
  - ...

OpenShift Application Pods

Agile

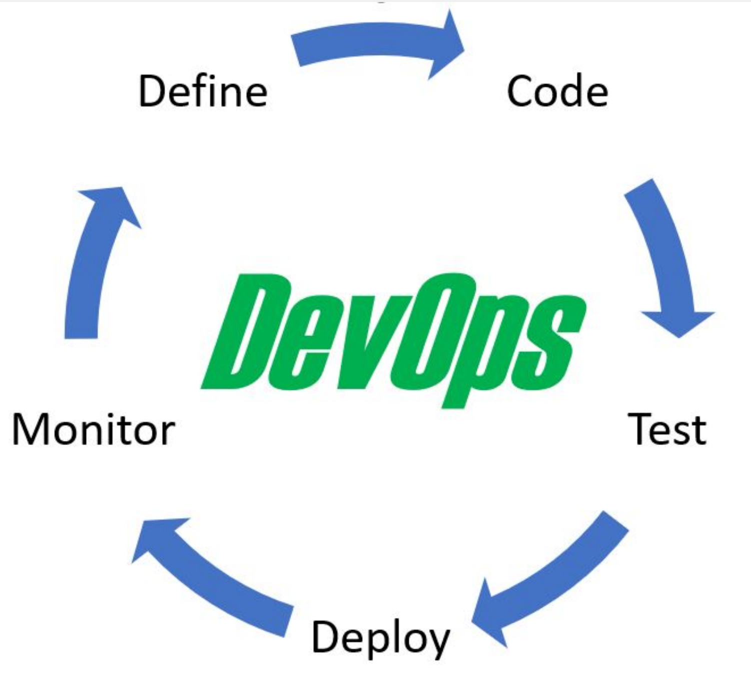
- Tomcat macro-services
- SQL: PostgreSQL (EDB) cluster
- NOSQL: MongoDB
- Pentaho (PDI)
- NGINX
- ...



CI-CD Pipeline

Agile

- Confluence -> Jira -> BitBucket
- Jenkins (CloudBees)
- Fortify
- JUnit
- Cucumber
- Maven
- Nexus
- WebInspect
- AppDetective
- ...



Outliers

Late to the party

- Containerizing
    - Splunk
    - MongoDB OpsManager
- Windows Dependencies
    - Selenium testing for IE and Edge browsers
    - Tableau

Lessons Learned

What we did right

- Hit delivery targets
- Tool selection
- Scrum for development
- “A” player mentality applied to hiring and retention
- Engaged DevOps from the beginning
- Communication, name selection and knowledge sharing

A+

“Failure is obvious, success is elusive”

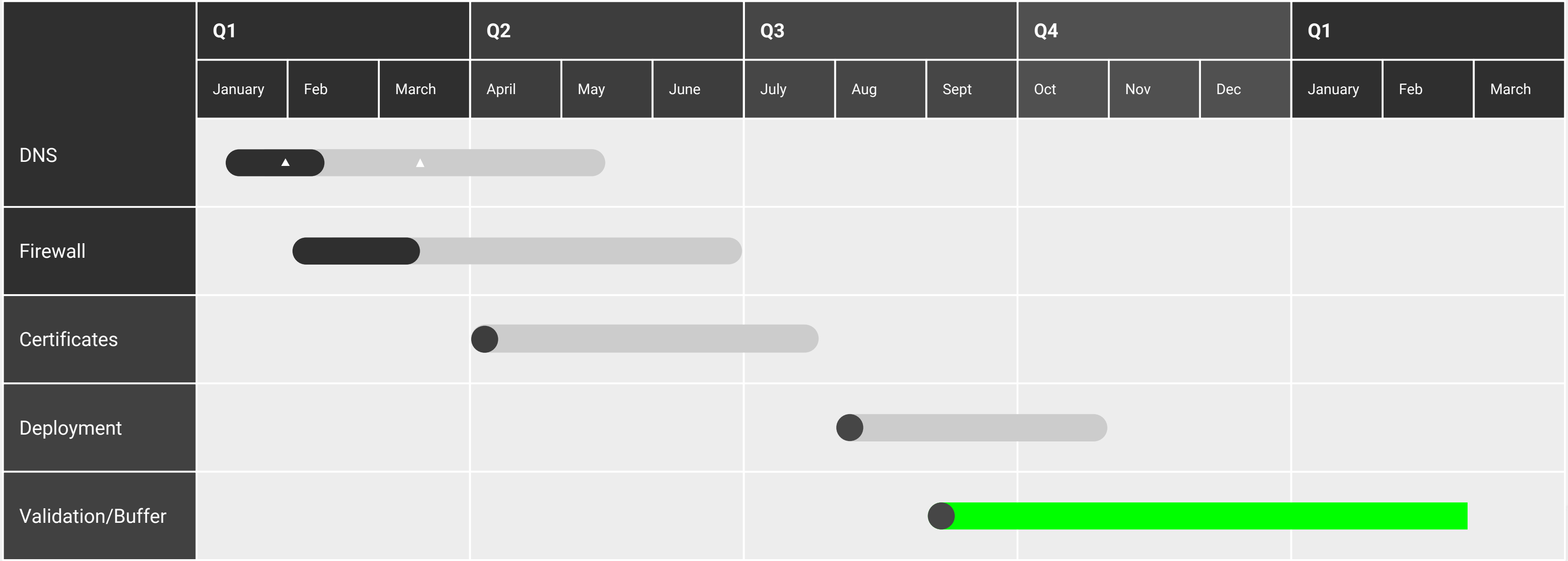
Lessons Learned

- Hit delivery targets…with hours to spare
- Scrum does not work for external dependencies + interrupt driven teams
- Underestimated DevOps team size
- Challenges running databases in containers
- Cut important components to meet deadlines
  - Monitoring build out
  - Live/Live
  - Load testing
- Coordination between multiple departments

"Make a lot of mistakes, just do not make the same mistake twice"

Creating Urgency

“Agile Fall” to prevent “Agile Fail”



Initial Delivery

Be a Challenger

But don’t blow it up

- We have the technology
  - Docker
  - Kubernetes
  - OpenShift

“Given time, money and great leadership anything is possible”

RED HAT SUMMIT

THANK YOU

linkedin.com/jerry.eshbaugh

twitter.com/eshbaugh