THE NEW PLATFORM: 
YOU AIN’T SEEN NOTHIN’ YET 

Gordon Haff 
@ghaff 
Technology Evangelist, Red Hat 
8 June 2016
ABOUT ME

Red Hat Technology Evangelist
Twitter: @ghaff
Podcast: Cloudy Chat
Newsletter: https://www.getrevue.co/profile/ghaff
Google+: Gordon Haff
Flickr: bitmason
Email: ghaff@redhat.com
Blog: http://bitmason.blogspot.com
Author: Computing Next
Former analyst and minicomputer/Unix guy
WE ARE IN A CAMBRIAN EXPLOSION
OPEN SOURCE INNOVATION DISRUPTING AS ARE ASSOCIATED PRINCIPLES AND PRACTICES
ENOUGH?
MUCH IS VERY NEW

2006

2007
SOME IS EVEN NEWER

2006

amazon web services

2007

2012

OPENSHIFT

2013

2014

kubernetes
COMPUTING AS PUNCTUATED EQUILIBRIA

- Rapidly changing environment
- Open source innovation and recombinations
- Intersecting trends
- Hard to predict
TRADITIONAL INFRASTRUCTURE & APPS

- Server, storage, & networking hardware
- Operating system
- Application & dependencies
- Application & dependencies
- Application & dependencies
(ONE OF) THE PROBLEMS

- Server, storage, & networking hardware
- Operating system
- Application & dependencies
ENTER HARDWARE VIRTUALIZATION

Server, storage, & networking hardware
with CPU, memory, I/O virtualization assists
WHAT CHANGED REALLY?

- Application components still installed within OS (e.g. `yum install`)
- Applications still long-lived & stateful “pets”
- Server sprawl to virtual machine sprawl
WHICH WAS SORT OF THE POINT

• Improved server utilization reduced CAPEX
• Without (initially) operational impact
JUST PULL COMPUTING OFF THE “GRID”?
THE DISCONTINUITY HITS

• “Software is eating the world”
• Digital transformation needs:
  • More effective software delivery
  • Reimagined componentized architectures
  • Scale
  • Pervasive sensors & access
NEEDED: A NEW PLATFORM FOR CLOUD-NATIVE APPS DEVELOPED USING DEVOPS
SOME APPLICATION PROBLEMS TODAY

• Coordination across large teams
• Brittle apps: minor changes cause major breakage
• Process bogged down by big deployments
• Different teams keep reinventing the wheel (in gratuitously different ways)
• Hard to scale
• Hard to experiment
CLOUD-NATIVE APPS

- “Small” independent components
- Accessed as a (micro)service through an exposed API
- Replaceable cogs (“ants”)
- Reliable at the service level
- Portable across hybrid infrastructures
DEVOPS APPLIES OPEN SOURCE TOOLS, PRINCIPLES, AND PRACTICES WITH:

• CULTURE of collaboration valuing openness and transparency

• Iterative AUTOMATION of process from development through ongoing operations

• An evolving PLATFORM that optimizes for flexible, dynamic workloads
DEVOPS + CLOUD = INDUSTRIALIZE
WHAT ARE THE ELEMENTS OF THIS FACTORY?

- Software-defined infrastructure
- Container ecosystem
- Orchestration and resource control
- Developer tooling
- Hybrid cloud management
FROM SERVERS TO DATACENTERS

- Software-defined
- Dynamic resource pool
- Physical/private/public/hybrid
OPENSTACK: SOFTWARE-DEFINED INFRASTRUCTURE
OPTIMIZED CONTAINERS: ISOLATION WITHIN OS

- Linux kernel manages isolation, resource use, and security
- Low overhead
- Bootable, immutable, versioned filesystem trees (OSTree)
- Namespaces
- SELinux
- Cgroups
MAKING CONTAINERS SECURE AND TRUSTED

- **ISOLATION OF HOSTS**: Host OS + SELinux maintained by trusted kernel engineers and frequently updated.
- **ARE SOURCES TRUSTED?**: 36% of Docker Hub official images contain high priority security vulnerabilities.*
- **WHAT’S INSIDE CONTAINERS**: Red Hat + Black Duck = secure, trusted model for validating container contents.
- **TRUST IS TEMPORAL**: New vulnerabilities are identified daily and containers become stale over time.

MAKING CONTAINERS USEFUL: ECOSYSTEM & DEFACTO STANDARDS

1 Open Container Initiative (OCI)
2 Cloud Native Computing Foundation (CNCF)
OPERATE AT SCALE: EVERYONE IS SCALING!

- Not just unicorns and mammoths
- Different aspects of scale:
  - Large scale workloads
  - Diverse workloads (batch and services)
  - Complex resource management (QoS, latency sensitivity, etc.)
- Grid computing: It lives!
- Orchestration and resource management
ROUNDING OUT THE DEV AND OPS EXPERIENCE

DEV

- Developer experience and tools
- Application lifecycle support (CICD, etc.)
- Service catalog
- xPaaS services

OPS

- Build platform and “get out of the way”
- Container management
- Hybrid cloud management
- Policy-based automation
- Audit support
THE NEW PLATFORM

Hybrid cloud management

Container resource management

Container-optimized Linux
Software-defined infrastructure
Container runtime & tools

Physical hardware

Developer tooling

Container/services
Container/services
Container/services
Container/services
Container/services

Physical hardware
THE CAMBRIAN EXPLOSION ISN’T OVER

- Role of hardware virtualization
- On-premise vs. public cloud trends
- Monoliths vs. Microservices
- Special-purpose hardware
- Balancing innovation with stability
The future is already here—it's just not very evenly distributed.

William Gibson
CREDITS

Cambrian explosion: Yale University
Punctuated equilibrium: University of California at Berkeley
Mainframe: "IBM 704 mainframe" by Lawrence Livermore National Laboratory. Licensed under Attribution via Commons - https://commons.wikimedia.org/wiki/File:IBM_704_mainframe.gif#/media/File:IBM_704_mainframe.gif
Ants: https://www.flickr.com/photos/pondapple/6502194585 Flickr Creative Commons license
Meteor, galaxy: NASA
Datacenter: Google
Dogs: https://www.flickr.com/photos/ulster/3250246355 Flickr Creative Commons license
Auto factory: Copyright Tesla
Tower: Daniel Pratts CC/flickr https://flic.kr/p/7RE6yc
Cambrian era: BBC
THANK YOU