

Containerized software for a modern world. The good, the bad and the ugly

Michael Waite

Red Hat

Technology Disruptions: Then and Now

2002

- Many Linux distributions
- Exciting technology but customers unclear: community powered
- Oracle, Intel and Red Hat align
- Customers benefit: improved performance, reduced costs, secure and supportable solutions

2019:

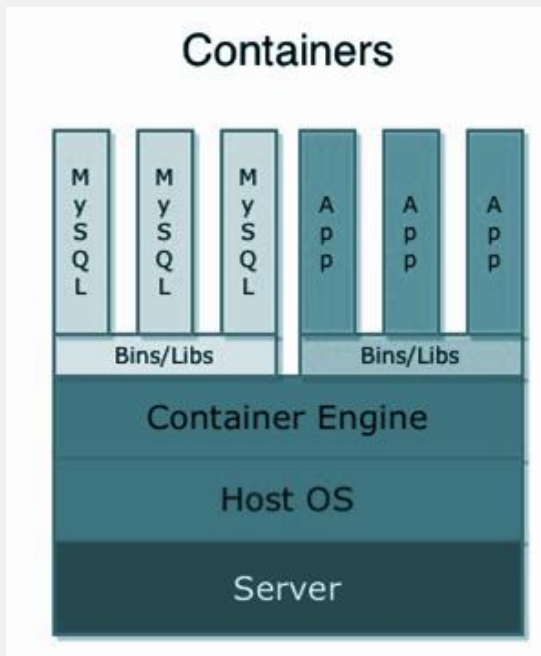
- Many Kubernetes vendors
- Exciting technology but customers unclear: community powered
- The ecosystem is forming
- Customers benefit: reduced costs, improved mobility, flexible capacity, secure and supportable

It's All About The Applications



- More apps = platform success
- Network effect drives value for application vendors & the platform
- Red Hat has a proven track record of making open source: secure, supportable and consumable.

Application Containerization for a Modern World



- Who is responsible for defect repairs within the code?
- Software applications and the underlying platform must be secure and supportable
- How is license management administered?
- How are support cases managed?

The Good

Easily deployed & consistent regardless of target environment.

Provides clean separation of concerns



The Bad

All of the compelling traits of containers result in software vendors becoming Linux maintainers



The Ugly

The promise of Microservices, the challenge of Containers

True or false?:

Security vulnerabilities will take care of themselves



Meltdown



Spectre



Key Takeaways

- Containers are a combination of Linux and software applications
- Customers need to know that they have a secure and supportable solution
- Red Hat partners with companies like Atomic Corp
- Joint testing and certification of the resulting containers
- Atomic Corp specializes in workload security. Red Hat specializes in the platform
- 2019 is realized and customers adopt microservices and cloud native apps for production use en-mass

Universal Truths

1. Water is wet.
2. The sky is blue



How Developers See Operations



How Operations Sees Developers



DevOps: Managing Application Containers

Scheduling: Decide where to deploy containers

Lifecycle: and health: Keep containers running despite failures

Discovery: Find other containers on the network

Monitoring: Visibility into running containers

Security: Control who can do what

Scaling: Scale containers up and down

Persistence: Survive data beyond container lifecycle

Aggregation: Compose apps from multiple containers



kubernetes

Raise Your Hand

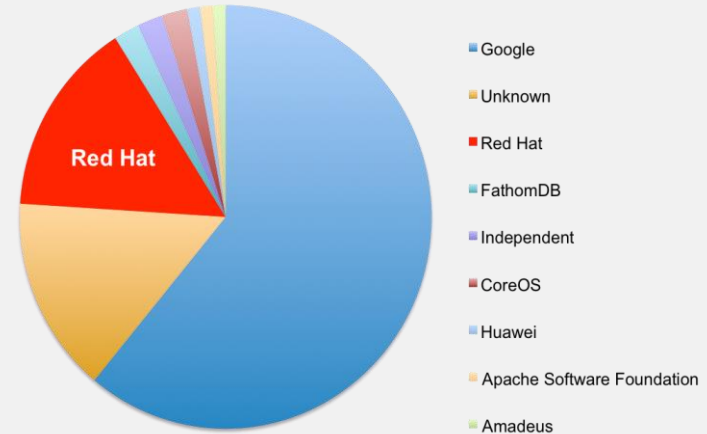
If you're a developer and you want to configure everything for each application you build.



If you're Operations and you want to manually oversee every application your developers build

Gaining Developer and Customer Confidence

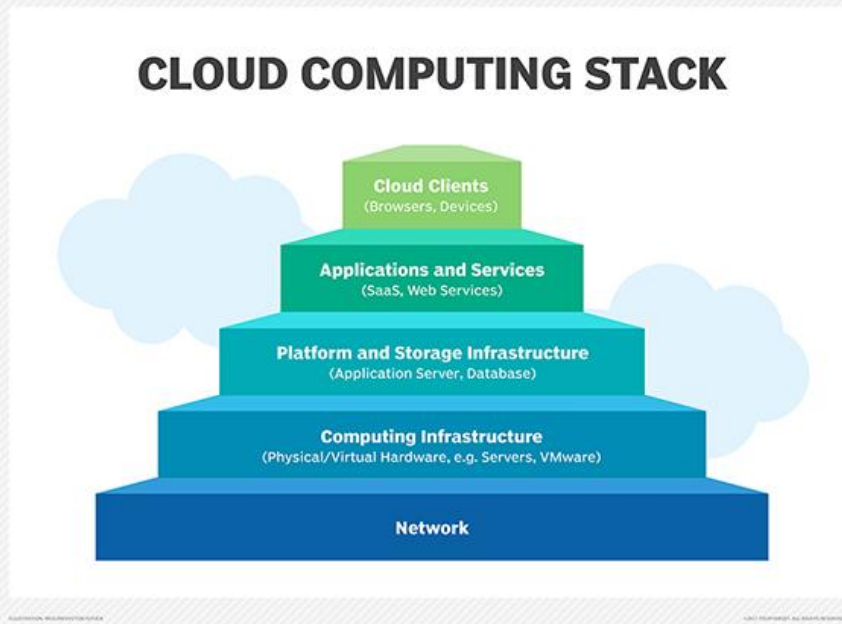
- Red Hat works with hundreds of software companies to test their applications on the Red Hat OpenShift platform
- OpenShift has the most tested, secure and certified software applications available
- The large ecosystem of applications and tools drives the network effect which benefits everyone



Kubernetes contributions

True or False?

The software application ecosystem is perfect and doesn't take much work or cooperation between platforms and ISVs.



Red Hat Connect - For Software Partners

Creating the world's largest application
container ecosystem.

Giving confidence to customers that our partners'
container-ized products are supportable for production use.



Thank You

