DevOps, Docker and PaaS with Red Hat OpenShift
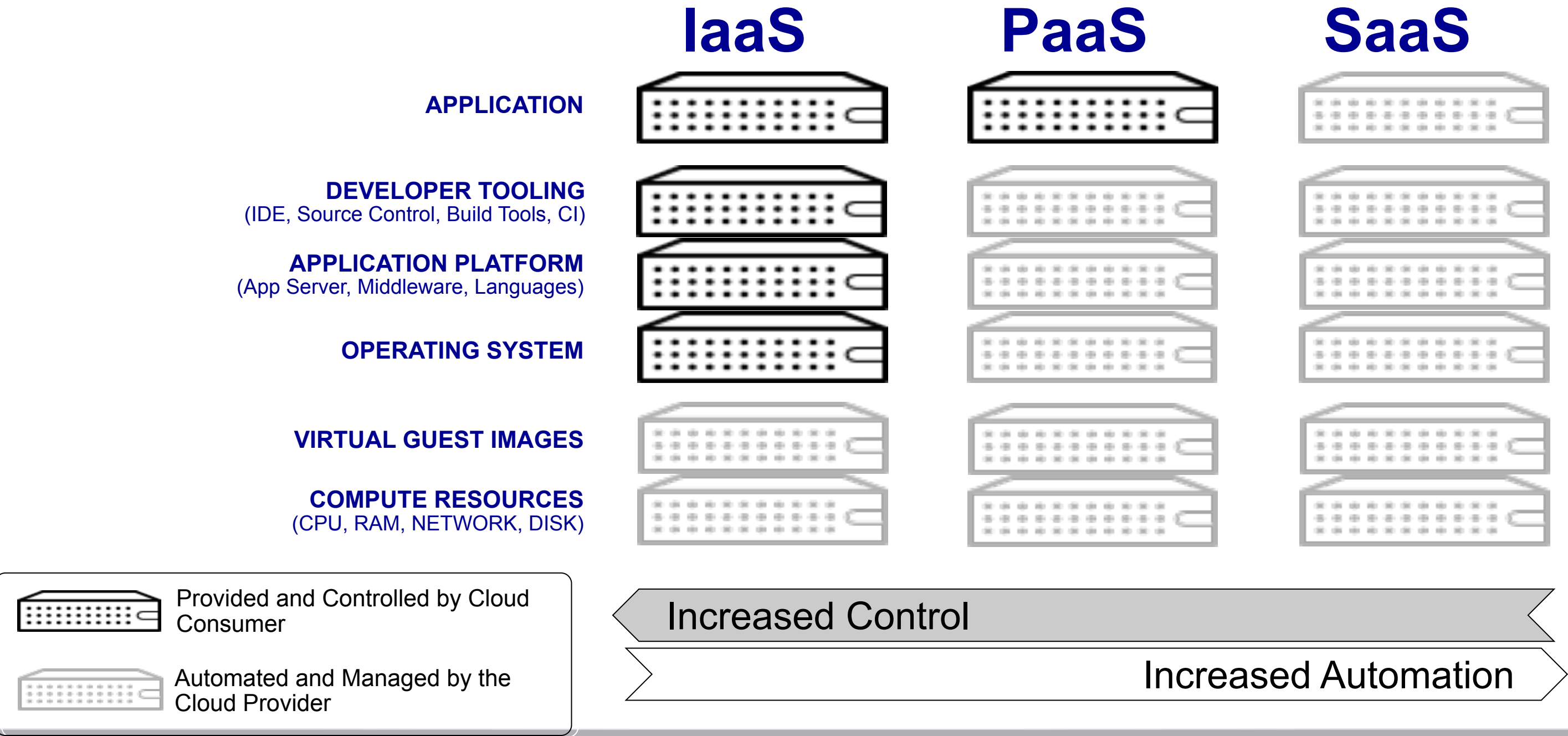
Introduction for the IRS
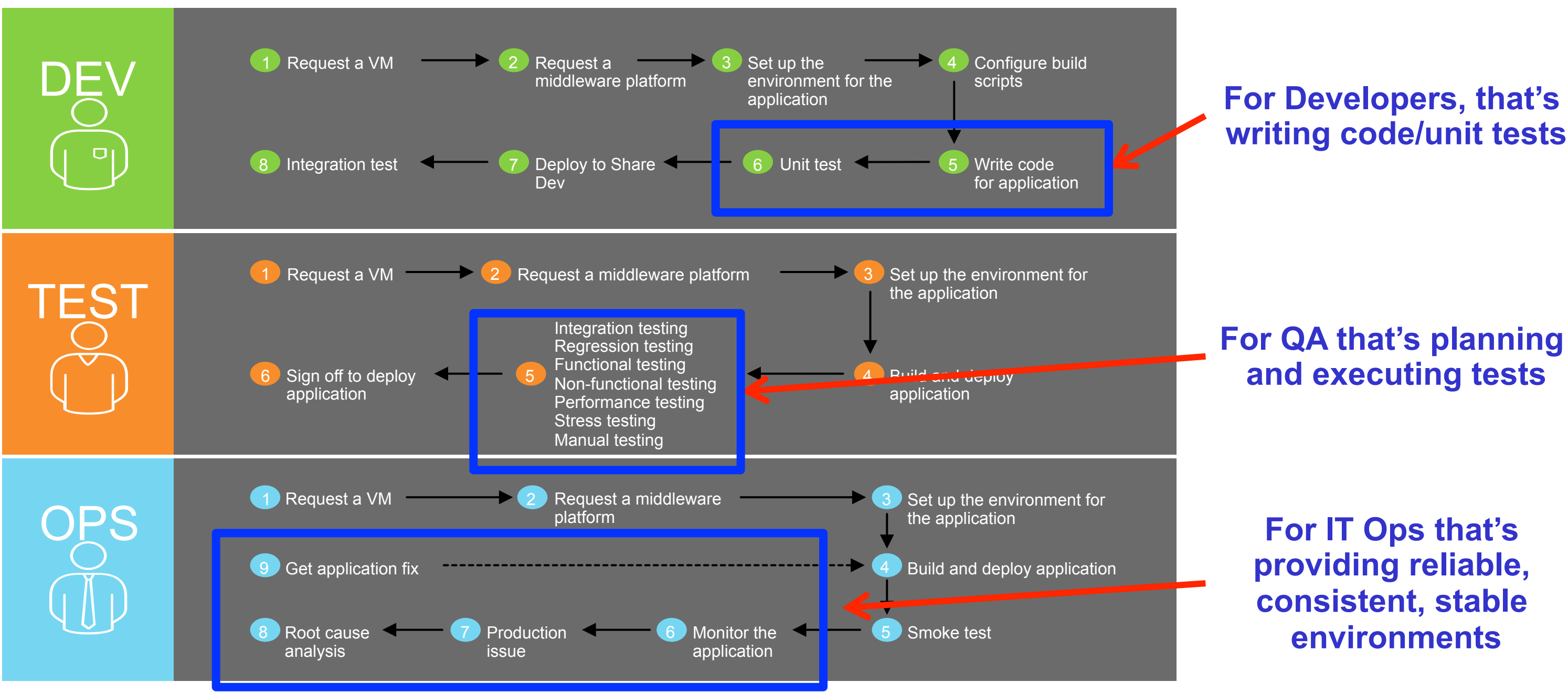
Red Hat Federal

September, 2015

PaaS Enables DevOps

IaaS **PaaS** **SaaS**

APPLICATION

DEVELOPER TOOLING
(IDE, Source Control, Build Tools, CI)

APPLICATION PLATFORM
(App Server, Middleware, Languages)

OPERATING SYSTEM

VIRTUAL GUEST IMAGES

COMPUTE RESOURCES
(CPU, RAM, NETWORK, DISK)

Provided and Controlled by Cloud Consumer

Automated and Managed by the Cloud Provider

Increased Control

Increased Automation

OPENSHIFT by Red Hat   redhat.

With PaaS, Skilled Resources Deliver Value

DEV

1. Request a VM → 2. Request a middleware platform → 3. Set up the environment for the application → 4. Configure build scripts

8. Integration test ← 7. Deploy to Share Dev ← 6. Unit test ← 5. Write code for application

For Developers, that's writing code/unit tests

TEST

1. Request a VM → 2. Request a middleware platform → 3. Set up the environment for the application

6. Sign off to deploy application ← 5. Integration testing Regression testing Functional testing Non-functional testing Performance testing Stress testing Manual testing ← 4. Build and deploy application

For QA that's planning and executing tests

OPS

1. Request a VM → 2. Request a middleware platform → 3. Set up the environment for the application

9. Get application fix ----→ 4. Build and deploy application

8. Root cause analysis ← 7. Production issue ← 6. Monitor the application ← 5. Smoke test

For IT Ops that's providing reliable, consistent, stable environments

OPENSHIFT by Red Hat   redhat.

Docker is a Great Enabler for PaaS

- **Standardization** – start every application from the same baseline configuration
  - ✓ But lots of choice – every vendor/project is creating Docker images

- **Speed** and **Simplicity** building and deploying with immutable images
  - ✓ Build once, deploy everywhere (and roll back easily)
  - ✓ Reduce post-deploy setup

- **Isolation** and **Density** simultaneously
  - ✓ Run multiple 10s to 100s of apps per host

- **Portability** from host to host, and environment to environment
  - ✓ Run on bare metal, VM, or Cloud (like AMI) – all you need is Linux

- Built for modern architectures – **Microservices**
  - ✓ Encourages creating small, interconnected components

OPENSHIFT
by Red Hat

redhat

Docker Alone is Not Enough for Enterprise Development

- Image-building is challenging, especially for complex applications
- Garbage collection – removal of old images (most end up writing by hand)
- Capturing logs from running Docker applications (again, roll your own)
- Security risks – privileged container processes can break out
- Custom effort to support stateful applications, connect storage   [1]
- Requires automation to build, deploy, promote, remove/replace
- No way to manage container dependencies
- Managing container health – your Virtualization Management tools can't

   handle containers

Why Docker is Not Yet Succeeding Widely in Production
sirupsen.com
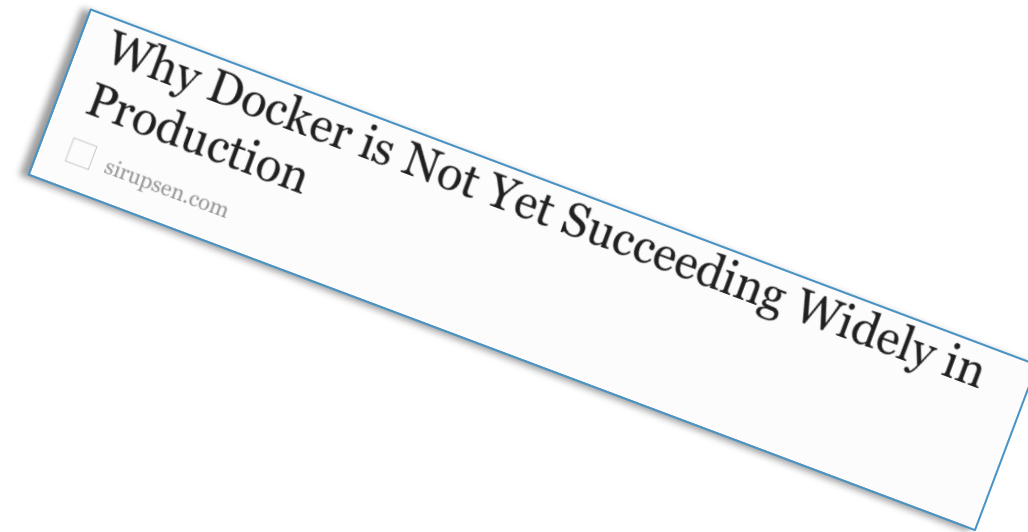
> *In the real world, individual containers fail a lot more than individual virtual machines…containers have to be run in managed clusters that are heavily scheduled and orchestrated. The environment has to detect a container failure and be prepared to replace it immediately. The environment has to make sure that containers are spread reasonably evenly across physical machines…and manage overall network and memory resources for the cluster. **It's a big job and well beyond the abilities of normal IT orchestration tools like Chef, Puppet, etc....**
>
> **- David Rensin, Google, "Kubernetes – Scheduling the Future at Cloud Scale"**

> *"Don't deploy Docker in production on your own infrastructure unless you are ready to take on the herculean task of configuring and managing applications in a containerized environment…**Don't build a PaaS with Docker**. Choose cloud PaaS and IaaS service providers that offer runtime, deployment, and orchestration of containers."*
>
> **Richard Watson, 4/25/2015 – "Gartner – Assessing Docker and Containers for Five Software Delivery Use Cases"**

[1] Simon Eskildsen - Shopify, July 2015, "Why Docker is not yet Succeeding in Production - http://sirupsen.com/production-docker/

OPENSHIFT
by Red Hat

redhat.

The Red Hat Solution - OpenShift

OpenShift Enterprise PaaS
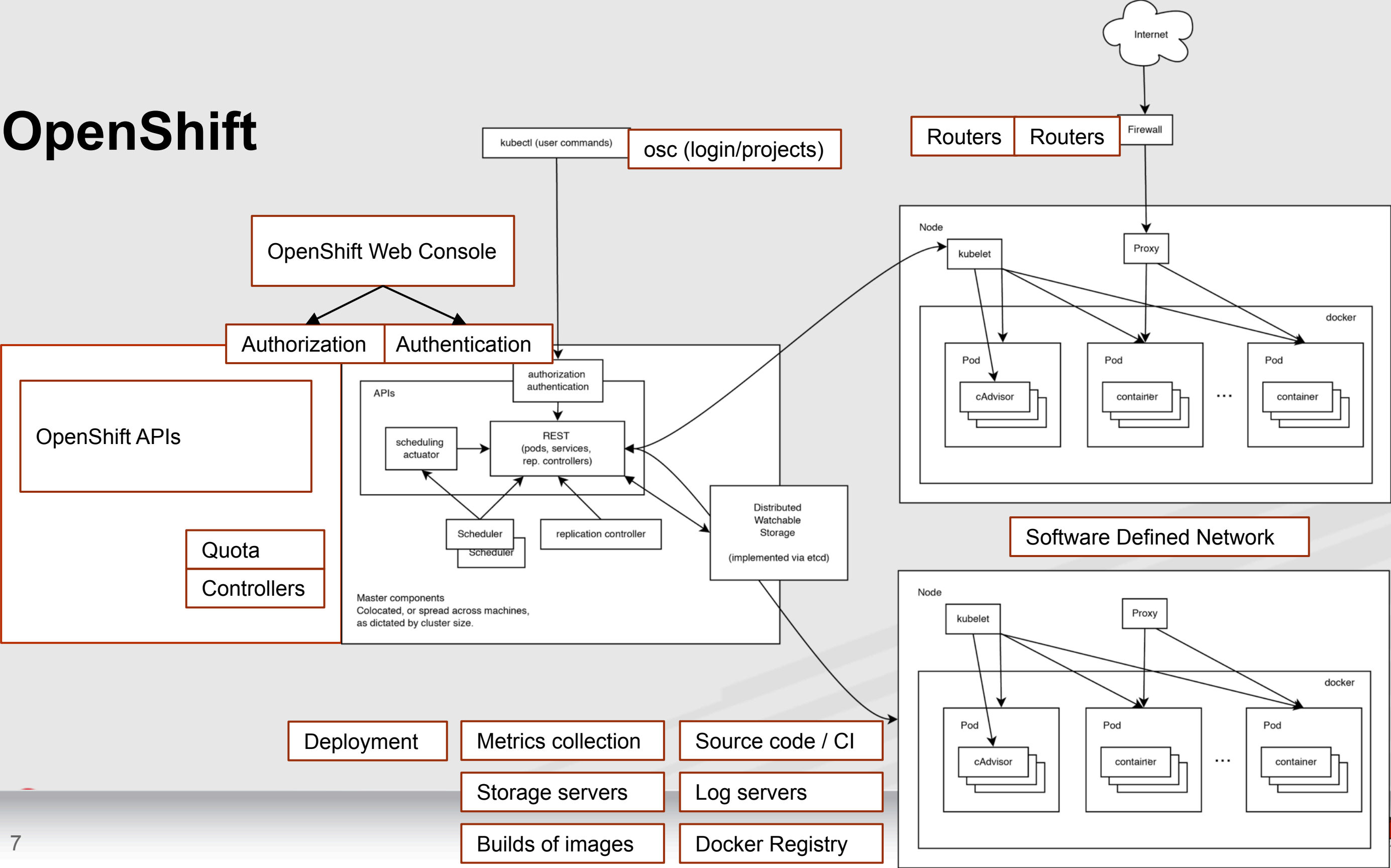
Docker	Kubernetes	OpenShift Origins
• Standardized, portable containers for any app, language, or framework	• Web-scale container orchestration with intelligent clustering and automated recovery	• Lifecycle of apps – build, deploy, manage, promote (CI/CD) • Manage thousands of applications and development teams • Manage users and quotas • Load balancing • Patching repository and running images

An OpenShift Subscription provides support for: Docker; Kubernetes; RHEL; Docker Repository; Out-of-Box Docker Images

OpenShift

osc (login/projects)

kubectl (user commands)

Routers   Routers   Firewall

Internet

OpenShift Web Console

Authorization   Authentication

OpenShift APIs

Quota

Controllers

APIs

authorization
authentication

scheduling
actuator

REST
(pods, services,
rep. controllers)

Scheduler
Scheduler

replication controller

Distributed
Watchable
Storage

(implemented via etcd)

Master components
Colocated, or spread across machines,
as dictated by cluster size.

Node

kubelet   Proxy

Pod
cAdvisor

Pod
container

Pod
container

docker

Software Defined Network

Node

kubelet   Proxy

Pod
cAdvisor

Pod
container

Pod
container

docker

Deployment   Metrics collection   Source code / CI

Storage servers   Log servers

Builds of images   Docker Registry

redhat.

Why Red Hat for Docker and Kubernetes

- Red Hat is the **only** company offering on-premise Support for Docker+Kubernetes

- Red Hat is the **only** company significantly contributing to both projects

  - ✓ #2 contributor to Docker, #2 contributor to Kubernetes

  - ✓ Red Hat's Clayton Coleman has changed more lines of Kubernetes code than any other developer

  - ✓ Red Hat has 46/237 Kubernetes developers, and 42/915 Docker developers (Docker, Inc. has 30)

  - ✓ Red Hat's Dan Walsh is on the Docker Governance Board

- RHEL+OpenShift delivers container security

- OpenShift completes the story: source-to-image; CI/CD; Team development; certified images; developer self-service; automated roll-back; image garbage collection; persistent storage connection; SDN; load balancing integration

Proven Value to the Enterprise with OpenShift

What I like about v3: *"So far, everything… this is pretty slick stuff. The rollback behavior in particular is something I've been trying to cobble together here for awhile."*

"With OpenShift we've built a push-button development stack" **– Ryan Granard – VP Platform Engineering - PayPal**
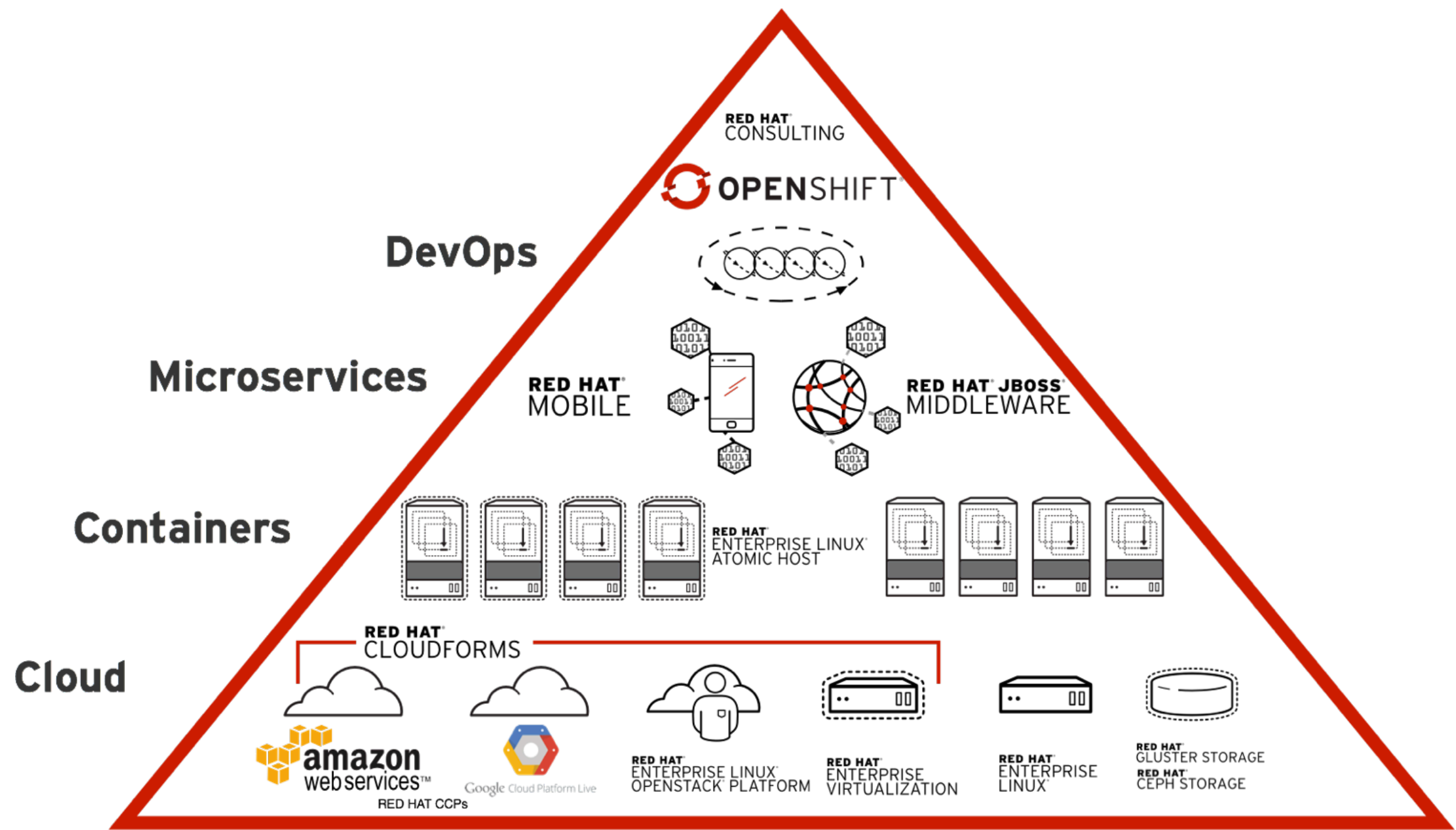
"Once we actually looked at and had all of the conversations… there was really only one choice and that was OpenShift. The only people that actually understood what it was that we were talking about was the Red Hat guys." **– Tony McGivern, CIO of FICO**

Because It's the only scalable application framework that fully delivers on all 4 key criteria: Capex; Opex; Isolation; and Flexibility. **– "Why Does CA Platform Use OpenShift" whitepaper.**

Red Hat Brings It All Together