

Docker, Cloud Foundry & OpenStack

Leading OpenSource Triumvirate - How do they all come together!

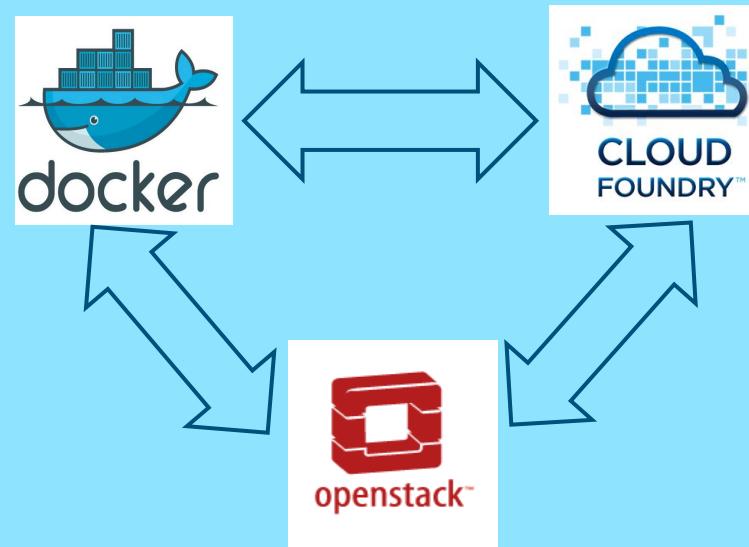
Animesh Singh @AnimeshSingh

IBM Cloud Architect

Accelerating Enterprise OpenStack



Docker, Cloud Foundry & OpenStack – Leading OpenSource Triumvirate!



Docker, Cloud Foundry & OpenStack - In Top Five !

Top 15 Open Source Cloud Computing Technologies 2014			
Position	Project	Categorie	Founded
1	OpenStack	Infrastructure	2010
2	Cloud Foundry	Platform	2011
3	KVM	Virtualization	2007
4	Docker	Virtualization	2013
5	Apache Mesos	Infrastructure	2012
6	MongoDB	Database	2009
7	Puppet	DevOps	2005
8	Chef	DevOps	2009
9	OpenShift	Platform	2011
10	Jenkins	DevOps	2011
11	Ceph	Storage	2011
12	Salt	DevOps	2011
13	CloudStack	Infrastructure	2010
14	CoreOS	Infrastructure	2014
15	CouchDB	Database	2005



<http://analystpov.com/cloud-computing/top-15-open-source-cloud-computing-technologies-2014-24727>



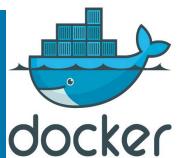
IBM cloud and open technologies

At all tiers, IBM is committed to building its cloud on an open cloud architecture

API
economy



Software
as a Service



Platform
as a Service



Infrastructure
as a Service

Products and services built on open source and open standards benefit IBM and customers



<http://www.ibm.com/developerworks/cloud/library/cl-open-architecture/>



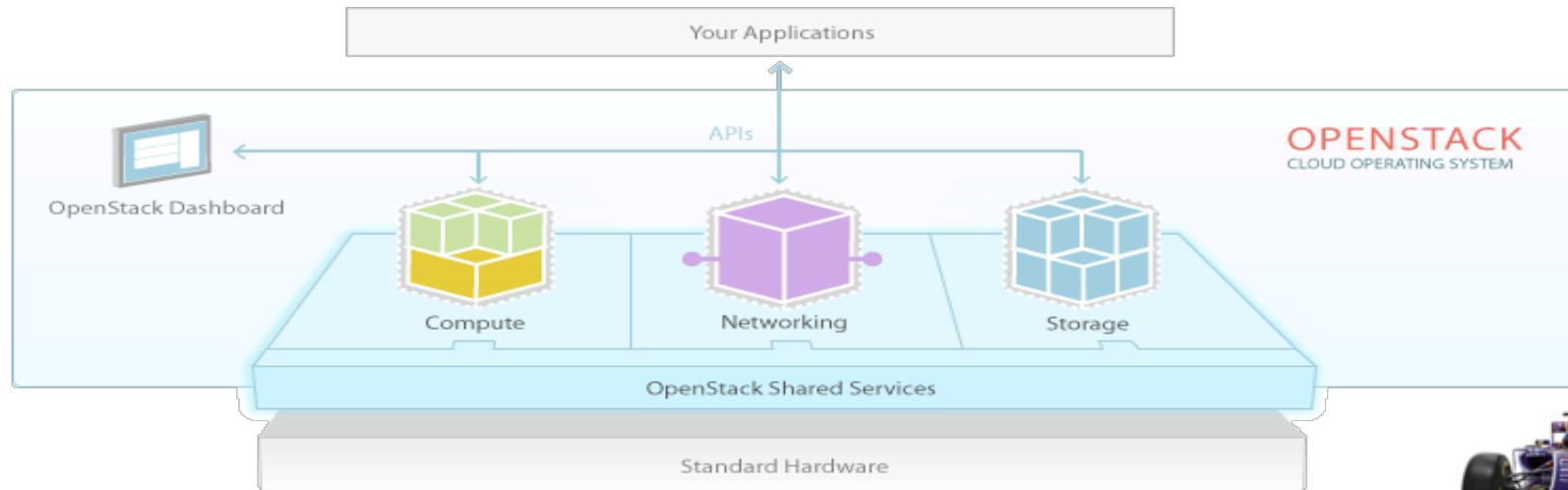
OpenStack



OpenStack Overview:

Collection of well integrated IaaS modules:

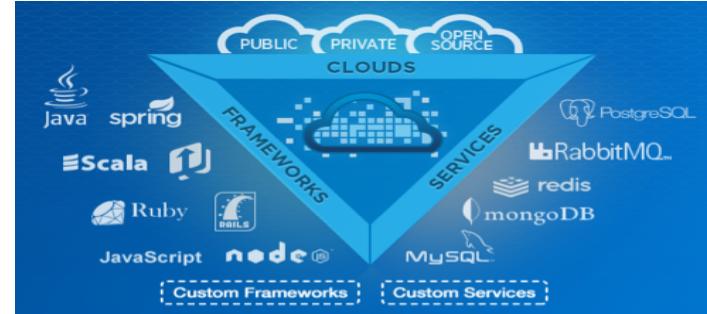
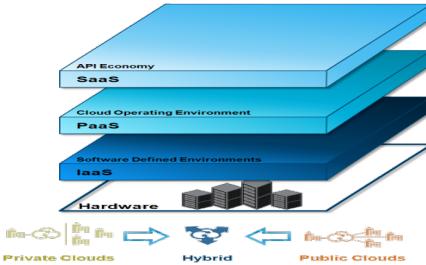
- Compute (Nova)
- Networking (Neutron)
- Object Storage (Swift)
- Block Storage (Cinder)
- Identity(Keystone)
- Image Service (Glance)
- Dashboard (Horizon)



Cloud Foundry



Cloud Foundry Overview



Open Cloud Platform

There is an increasing appetite for cloud-based mobile, social and analytics applications from line-of-business executives - drives the need for a more open cloud development platform

TOTAL CONTRIBUTORS
1,165

Meets Developer's Needs
Focus on app development, not provisioning VMs, databases, messaging servers, etc
Agile development model
Deploy and scale in seconds

Compelling Community

Cloud Foundry has a compelling community and emerging ecosystem as well as a mature set of capabilities and robustness

LINES OF CODE
739k

Platinum Founding Sponsors

Pivotal™

SAP

IBM

vmware

EMC²

hp



openstack™
CLOUD SOFTWARE

Cloud Foundry Developer Experience

CLI

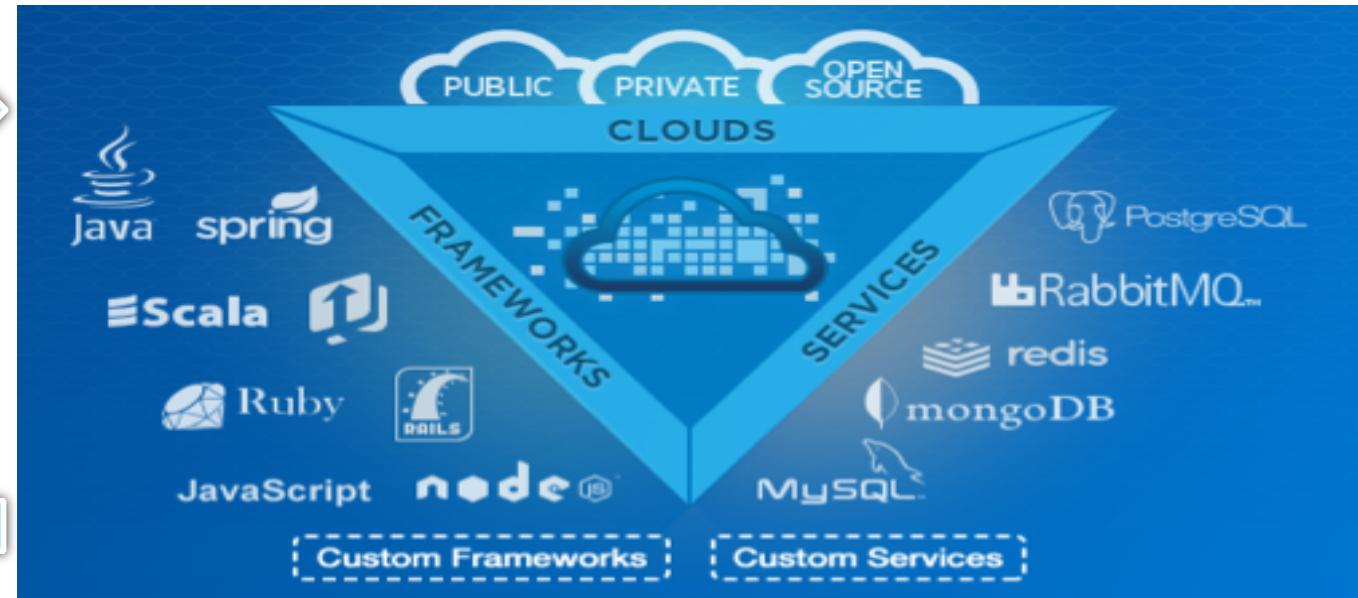
Eclipse IDE

Browser

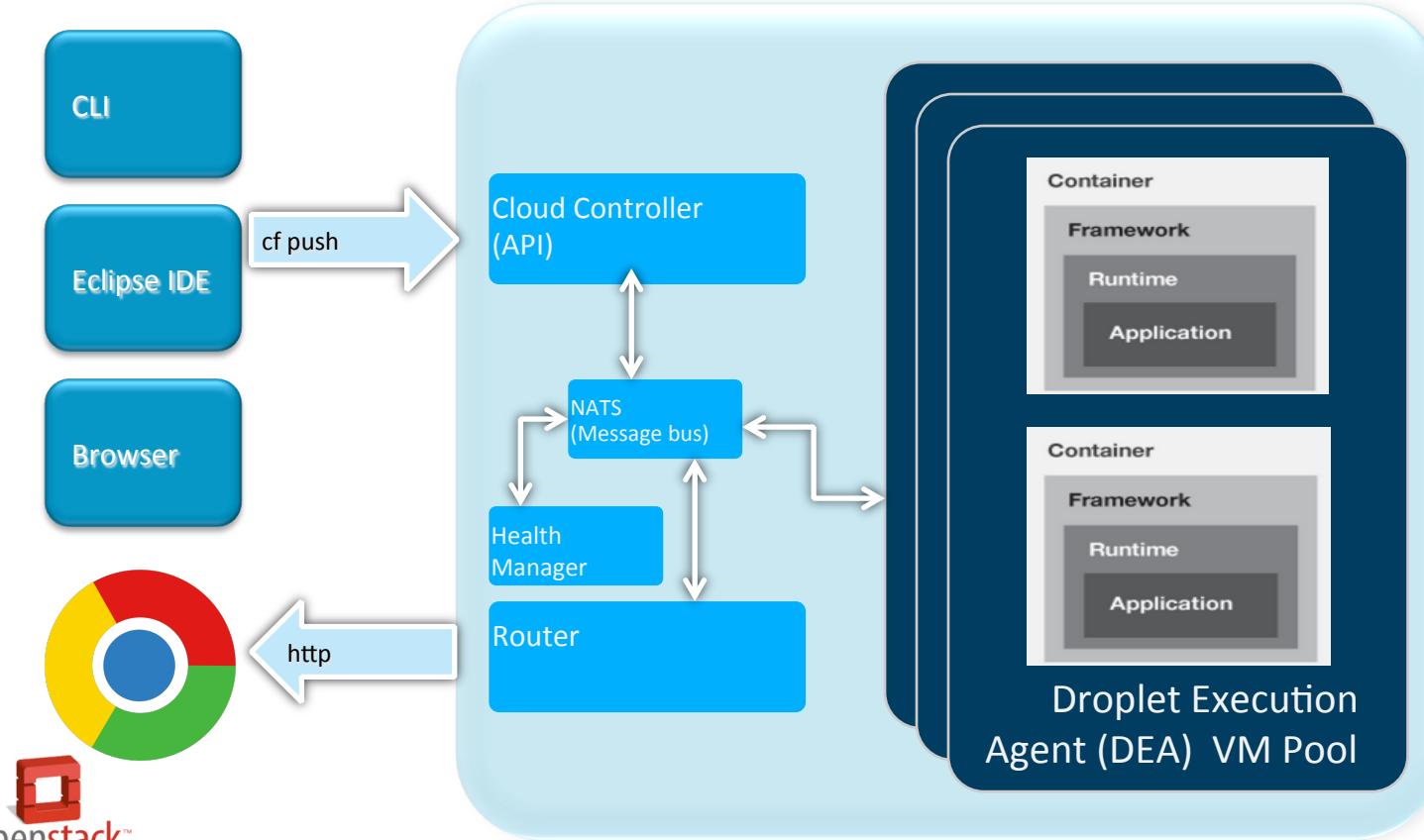
cf push



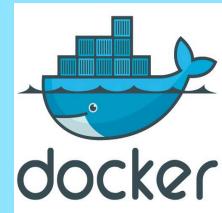
http



Cloud Foundry Architecture



Docker



Docker Overview

- ✓ One of the most disruptive technologies of recent past
- ✓ Every significant vendor (IBM, RedHat, Google, AWS, VMWare etc) has announced support for Docker
- ✓ First Docker conference was a huge success – with over 1000 attendees

What is it ? A tool to

- Run applications: An open source tool to run applications inside of a Linux container, a kind of light-weight virtual machine
- Package applications: In addition to running, it also offers tools to package containerized applications through Docker files
- Distribute applications: Create your own Docker registries or hubs, a cloud service for sharing applications and automating workflows.



openstack™



Google Cloud Platform

amazon
web services™

SOFTLAYER™

Rackspace.
the open cloud company

OPENSHIFT

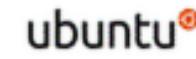
Microsoft Azure



redhat



CoreOS



ubuntu®



debian



fedora®

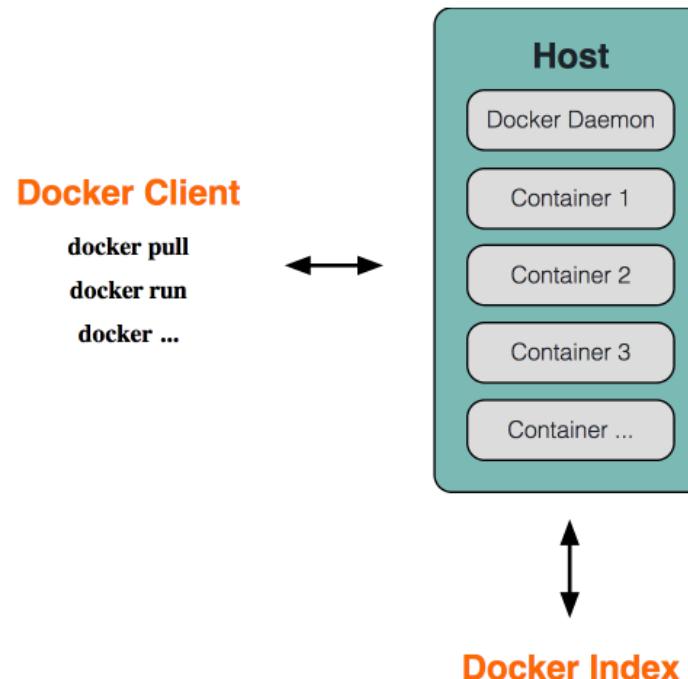


CentOS



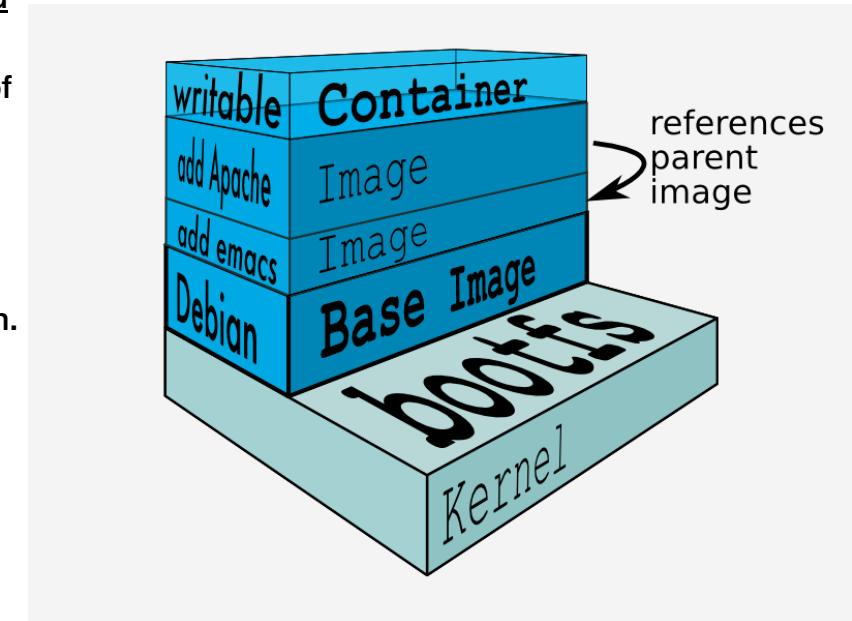
Docker Architectural Overview

- Docker uses a client-server architecture.
- The Docker client talks to the Docker daemon, which does the heavy lifting of building, running, and distributing your Docker containers.
- Both the Docker client and the daemon can run on the same system, or you can connect a Docker client to a remote Docker daemon.
- The Docker client and daemon communicate via sockets or through a RESTful API.



Docker Containers

- ✓ A Docker container consists of an operating system, user-added files, and meta-data – Basically a way to run mini operating systems in your host operating system with strong guarantees of isolation
- ✓ The Docker image is read-only. When Docker runs a container from an image, it adds a read-write layer on top of the image (using a union file system) in which your application can then run.
- ✓ Underlying Technology : Written in Go and makes use of several Linux kernel features
 - Namespaces - pid, net, mnt, ipc, etc.
 - Control Groups - cgroups (memory, cpu, blkio, devices)
 - Union File Systems - UnionFS (AUFS, btrfs, vfs)
 - Container Format - libcontainer or LXC



Docker Containers vs Virtual Machines

Virtual Machines

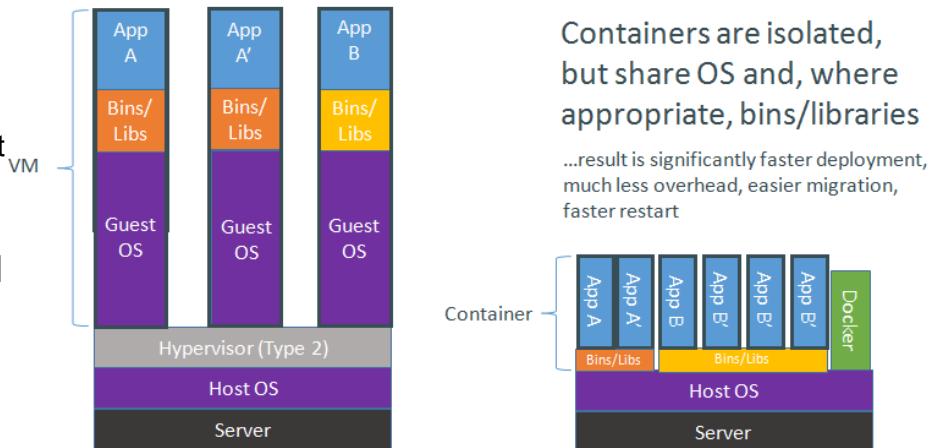
Each virtualized application includes not only the application - which may be only 10s of MB - and the necessary binaries and libraries, but also an entire guest operating system - which may weigh 10s of GB.

Docker

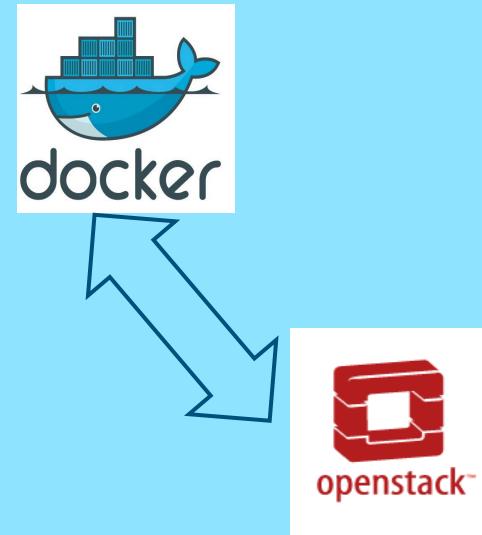
The Docker Engine container comprises just the application and its dependencies. It runs as an isolated process in userspace on the host operating system, sharing the kernel with other containers

Compared with Hypervisors, Docker which is OS-Level Virtualization:

- CPU Performance => native performance
- Memory Performance => few % for (optional) accounting
- Network Performance => small overhead; can be optimized to zero overhead
- creating a new base image takes a few seconds (copy-on-write)
- apps in different containers can share the same binaries / libs

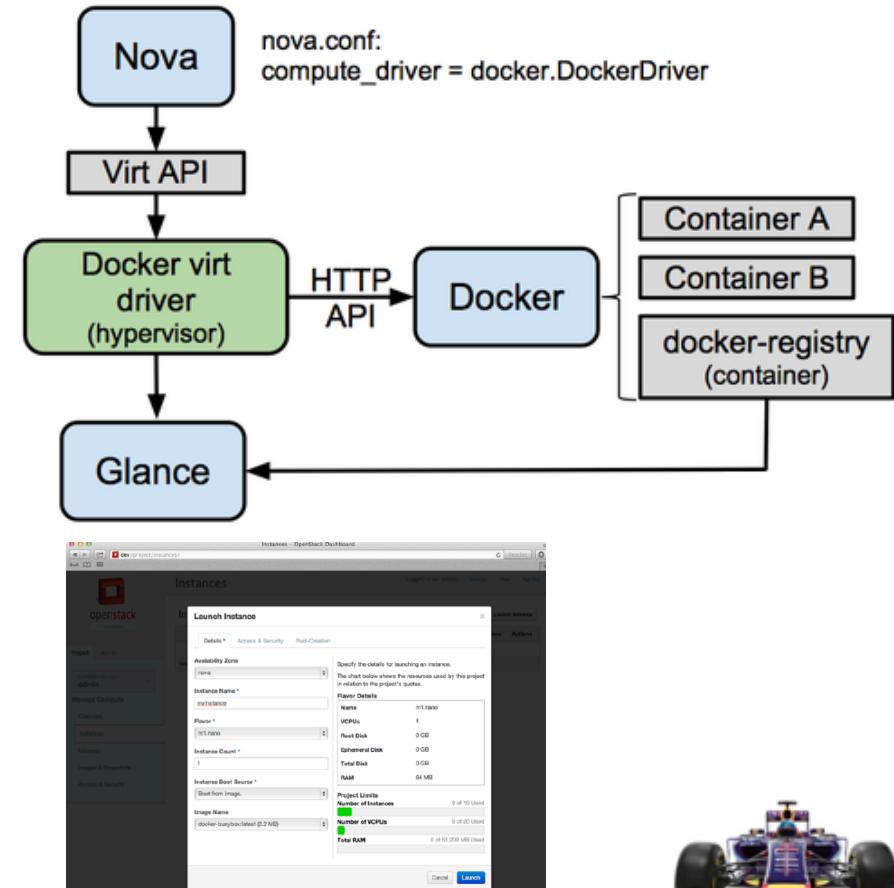


How do Docker and OpenStack intersect ?



Docker in OpenStack

- ✓ Docker is hypervisor driver for Openstack Nova Compute. It was introduced with the Havana release, but lives out-of-tree for Icehouse and Juno.
- ✓ The Nova driver embeds a tiny HTTP client which talks with the Docker internal Rest API through a unix socket. It uses the HTTP API to control containers and fetch information about them.
- ✓ The driver will fetch images from the OpenStack Image Service (Glance) and load them into the Docker filesystem. Images may be placed in Glance by exporting them from Docker using the 'docker save' command.
- ✓ Well integrated with Horizon UI

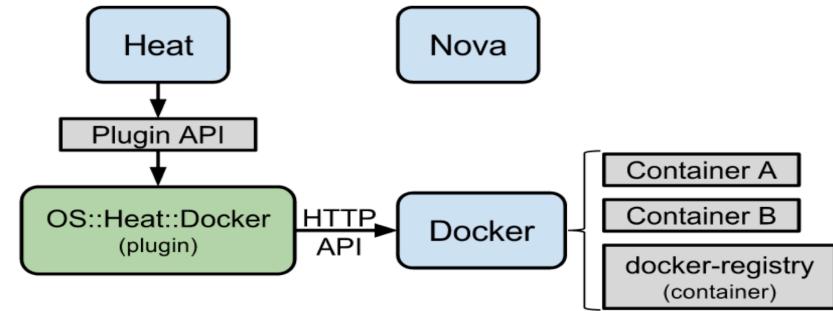


Source: <https://wiki.openstack.org/wiki/Docker>



Heat Template Plugin for Docker in OpenStack

- ✓ Not a replacement for Docker Nova Driver
- ✓ OpenStack Orchestration (Heat) is a solution for providing orchestration of resources inside OpenStack clouds.
- ✓ Using the Heat plugin, users may deploy and manage Docker Containers on top of traditional OpenStack deployments, making it compatible with existing OpenStack clouds.
- ✓ It allows to use the whole Docker API from a Heat template, and makes linking of containers easier
- ✓ In the example here, multiple containers may be created and linked together by simply adding more sections like "my_docker_container". They're not constrained by the OpenStack APIs and may leverage the full power of the Docker Remote API.



```
heat_template_version: 2013-05-23

description: Single compute instance running cirros in a Docker container.

resources:
  my_instance:
    type: OS::Nova::Server
    properties:
      key_name: ewindisch_key
      image: ubuntu-precise
      flavor: m1.large
      user_data: #include https://get.docker.io
  my_docker_container:
    type: DockerInc::Docker::Container
    docker_endpoint: { get_attr: [my_instance, first_address] }
    image: cirros
```



UI Integration with Horizon

Horizon UI > Orchestration > Stacks

Select Template

Template Source *

URL

Description:

Use one of the available template source options to specify the template to be used in creating this stack.

An external (HTTP) URL to load the template from.

Template URL

<https://raw.github.com/dotcloud/openstack-heat-d>

Cancel Next

Configure the Stack (parameters of the Heat template)

Launch Stack

Stack Name *

MyBlog

Description:

Name of the stack to create. If the provided values.

Creation Timeout (minutes) *

60

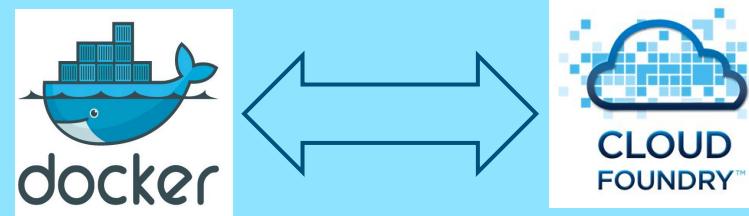
Rollback On Failure

Password for user "admin" *

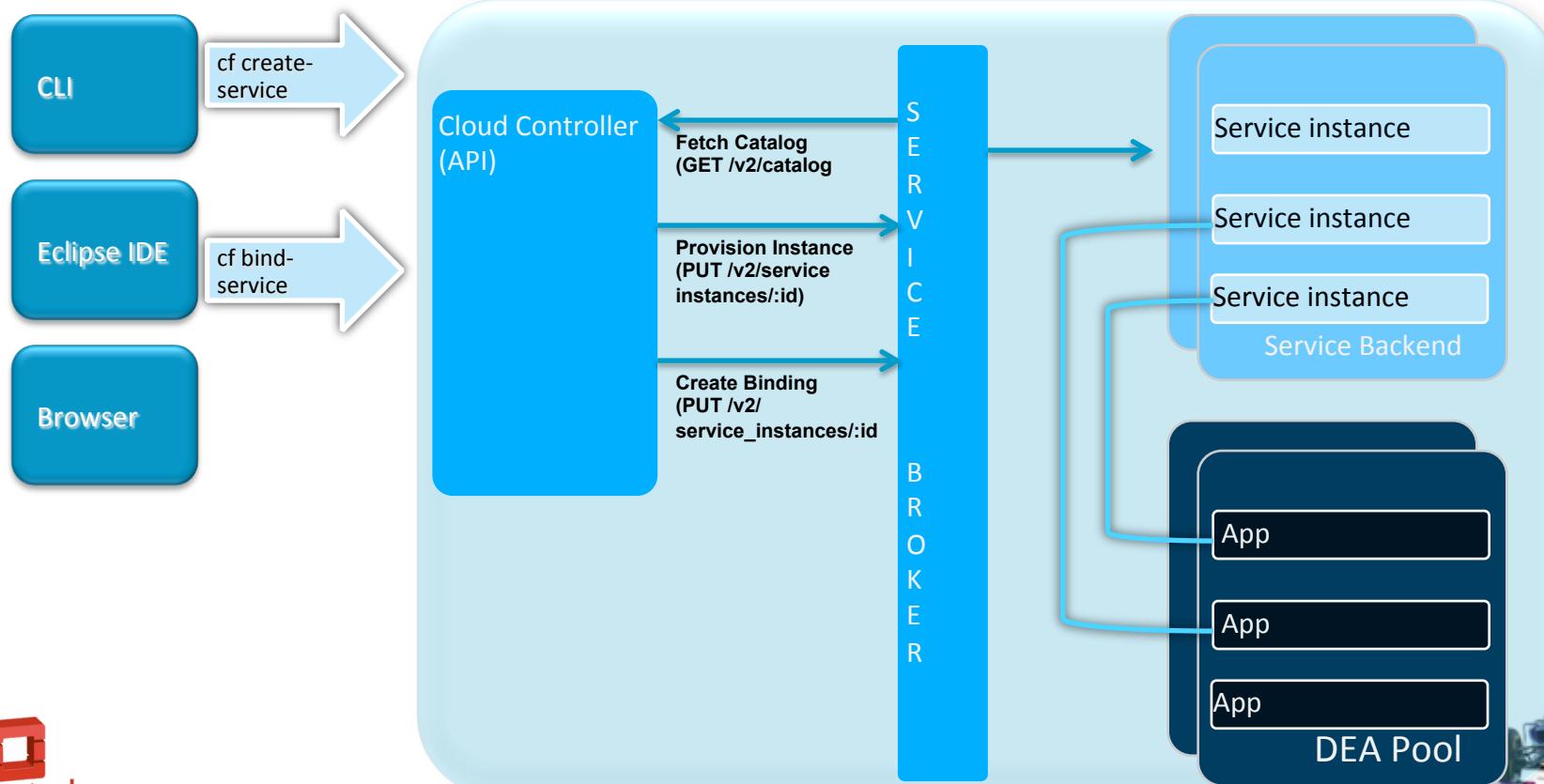
Cancel Launch



How do Docker and Cloud Foundry intersect ?



Cloud Foundry Service Broker



Docker Service Broker for Cloud Foundry

(Source : Ferran Rodenas, Pivotal)

An easy and convenient way to expose development and testing services to your applications without the overhead of creating an specific service broker by just using Docker images

- Services Catalog: predefined Docker services
- Provision an instance: create and start a predefined Docker container and assign random credentials via environment variables
- Bind an instance to an application: send service credentials hash back to the bound application
- Unbind an instance
- Unprovision an instance: destroy Docker container
- Expose a management dashboard: top processes, stdout/stderr logs, ...
- Syslog drain URL: drain your application logs to a Docker syslog (logstash, ...)



Diego – Rewrite of Cloud Foundry Runtime (supporting Docker)

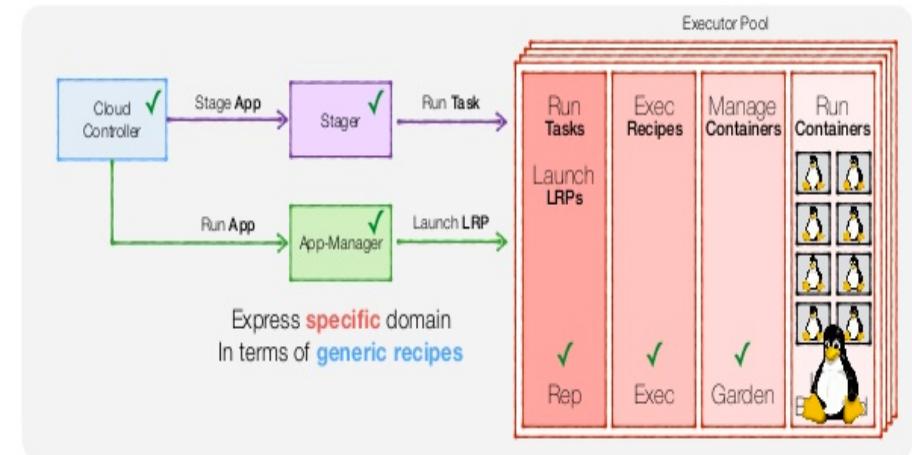
- ✓ Diego is a rethinking of the Droplet Execution Agent (DEA) within the Cloud Foundry Runtime. The primary functions of the DEA are to stage apps, run them in Warden containers, and manage their lifecycle by starting and stopping apps upon request of the Cloud Controller component.

- ✓ Number of problems in the current Elastic Runtime model:

1. Tight coupling
2. Poor separation of concerns
3. Creating “triangular dependencies”

- ✓ Cloud Foundry container technology, Warden has been rewritten in Go and is called Garden (now).

- ✓ Garden separates the server from backend, and it supports using Docker as root file system of a container.



Platform Independent ✓ →



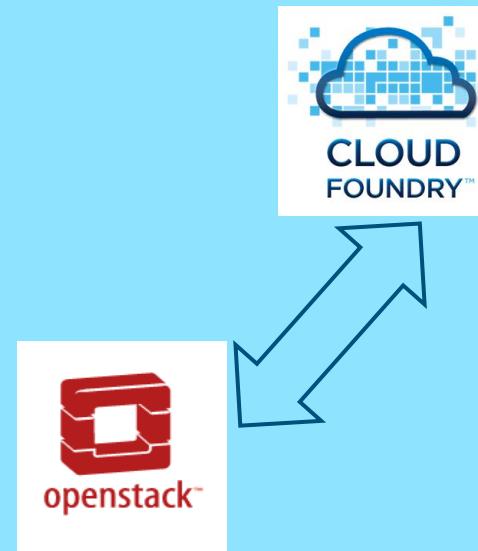
Cloud Foundry – Docker – Phase 1 Use Cases (Support being added)

Use Cases

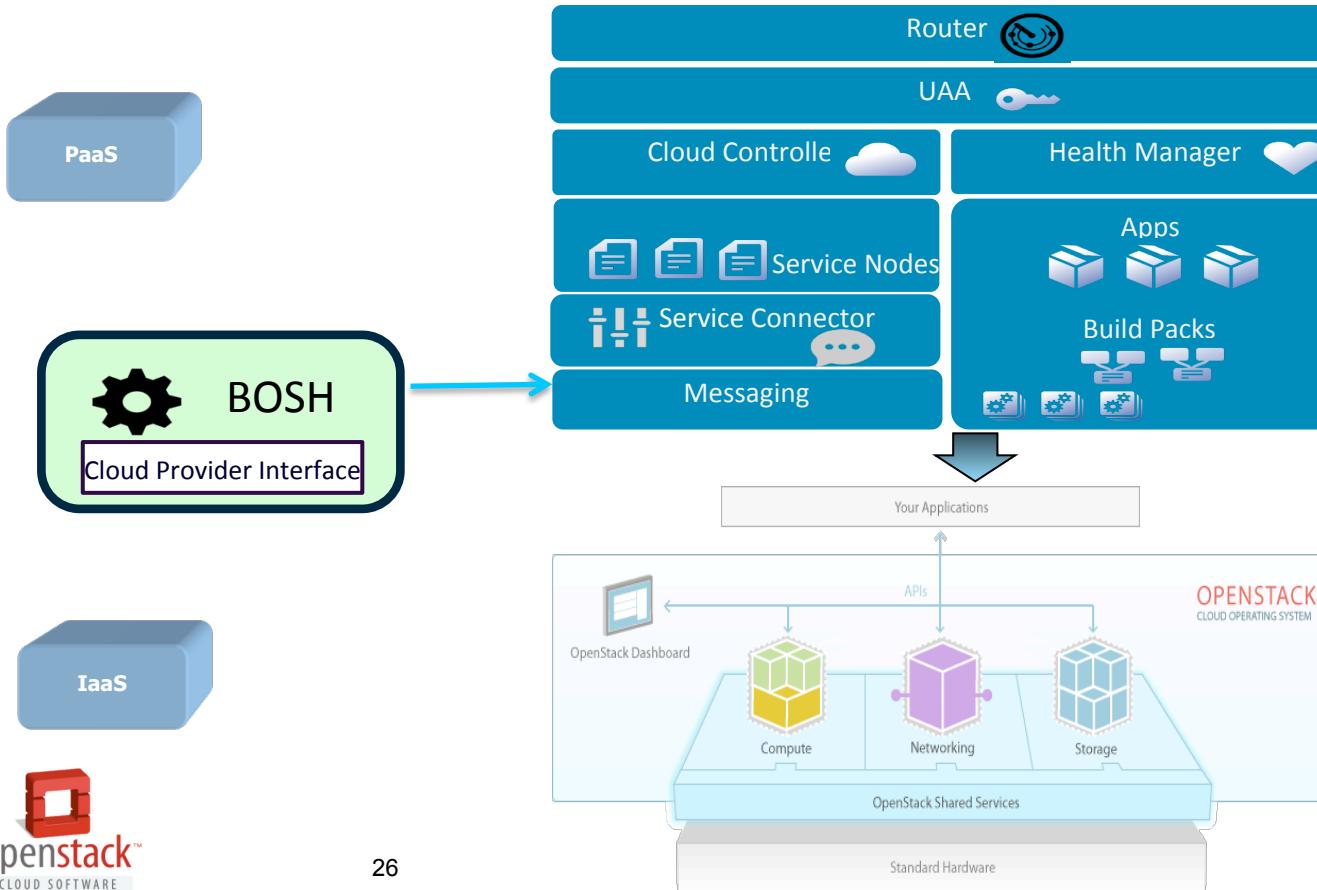
- As a user I want to push docker images from the public <https://hub.docker.com> (not Dockerfile)
- As a user I want to be able to start/stop and scale my docker image based application
- As a user I can see the health of my docker image based application
- As an operator I want to be able to toggle docker image support



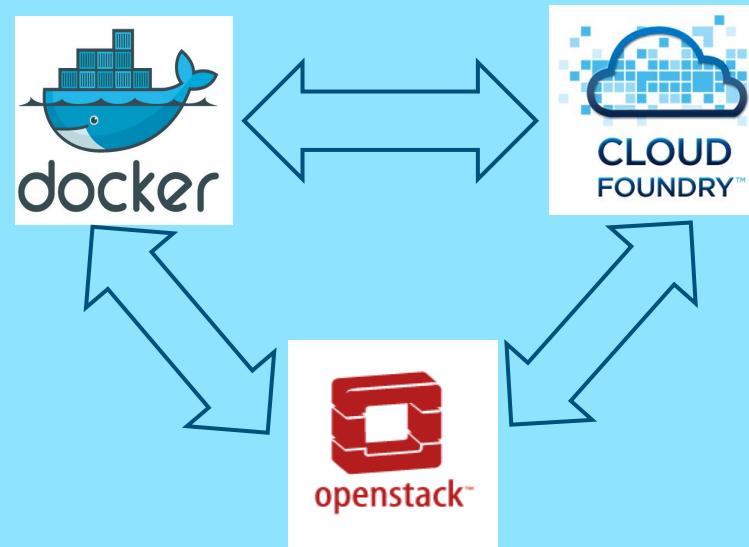
How do Cloud Foundry and OpenStack Intersect



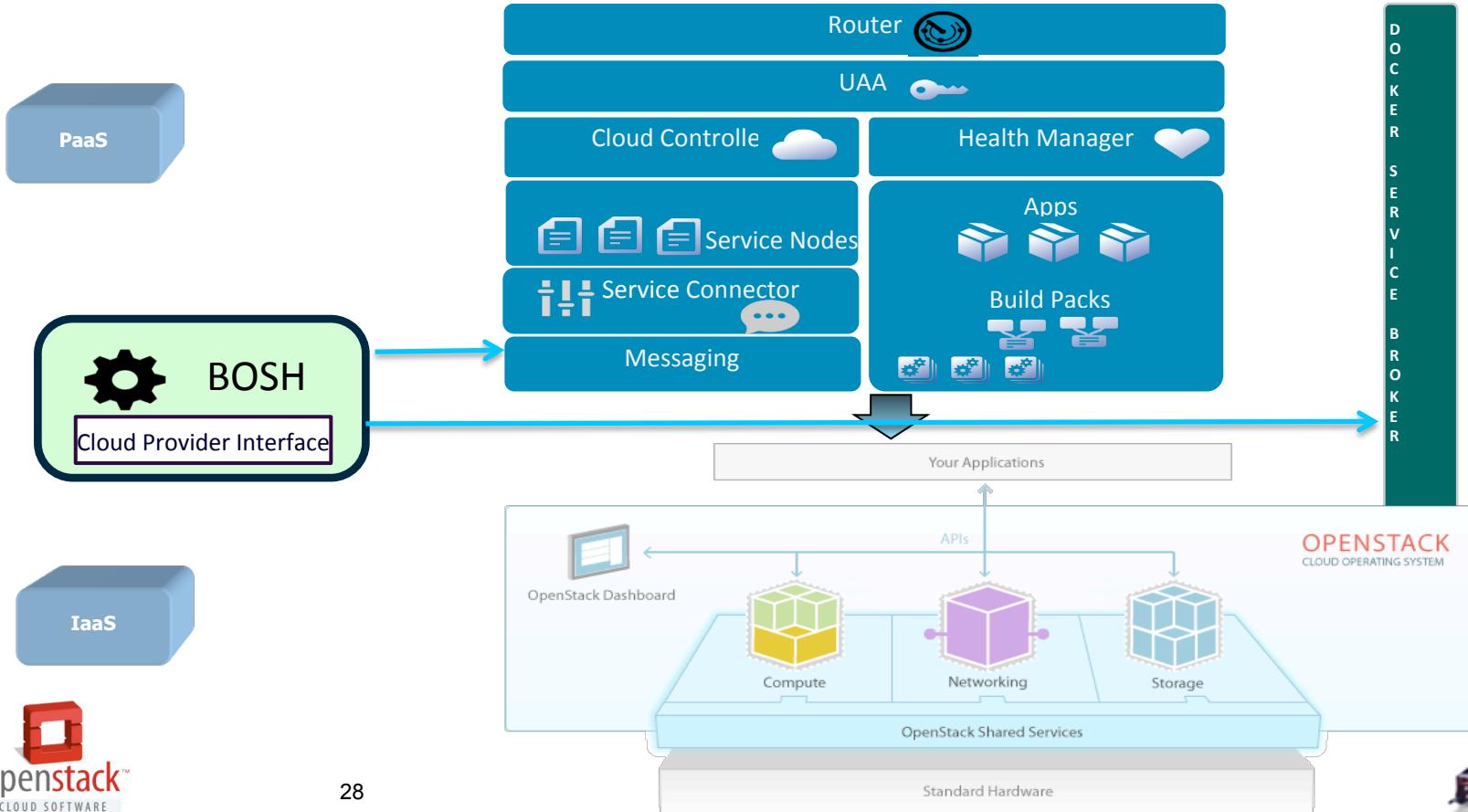
Cloud Foundry , OpenStack Integration



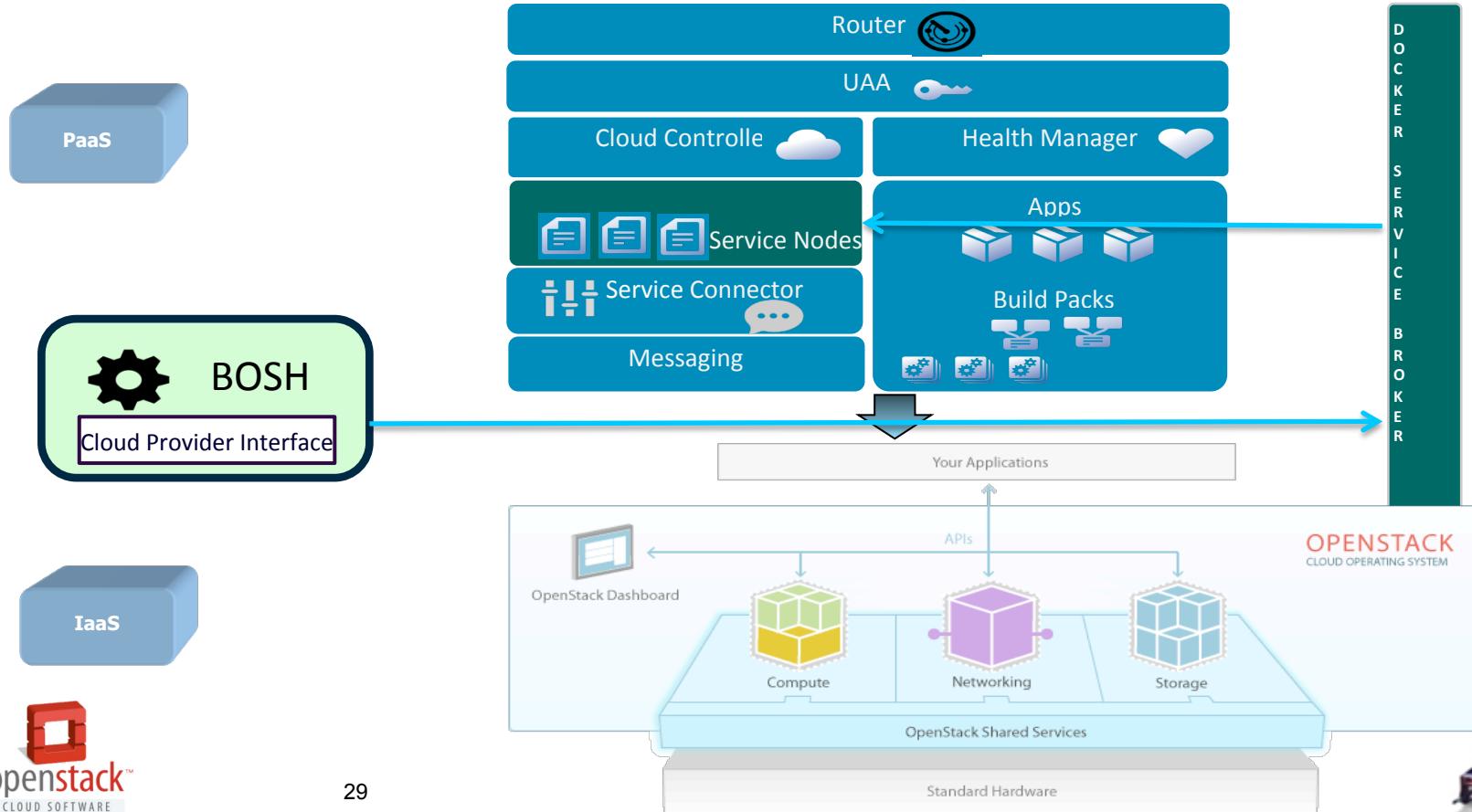
How do they all come together?



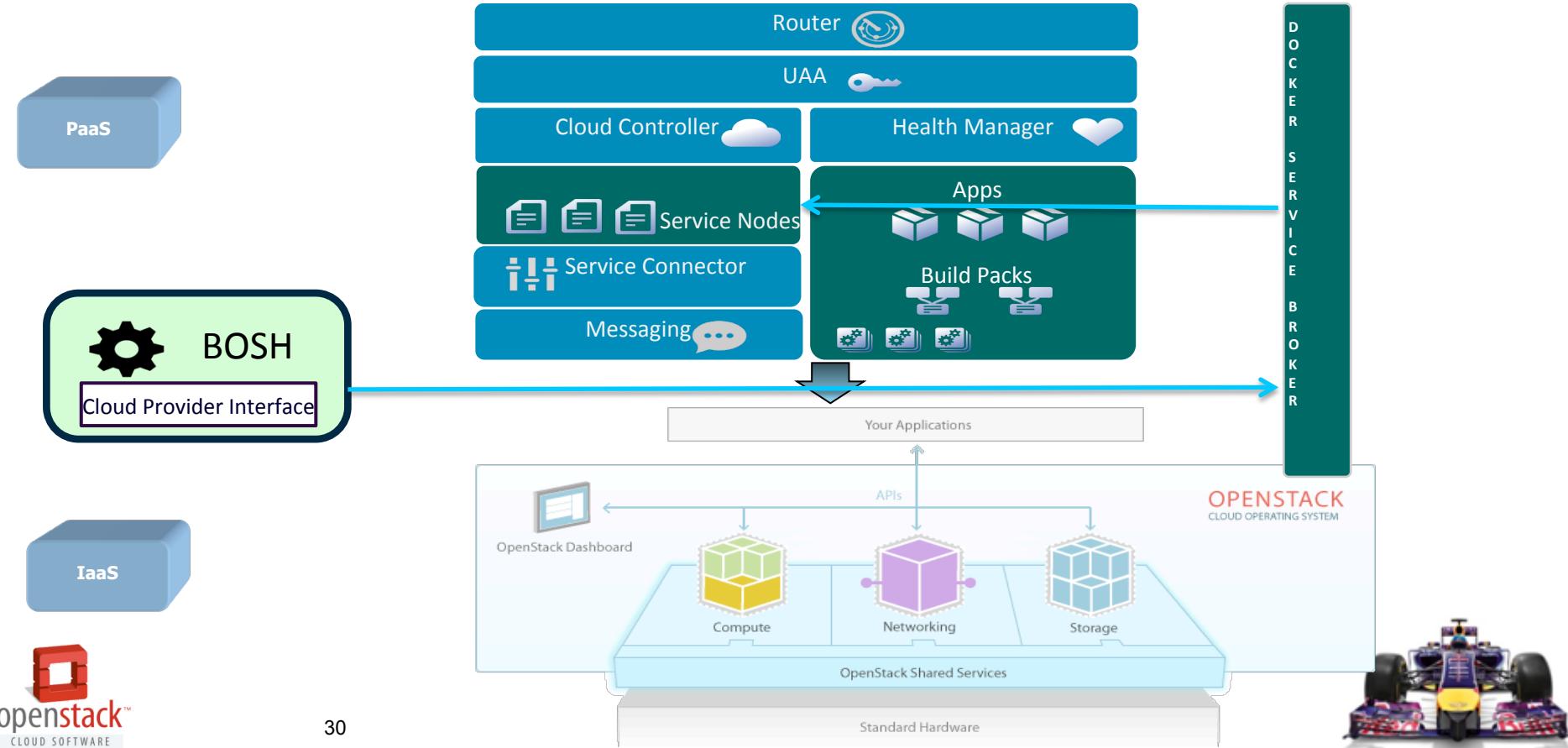
Cloud Foundry , OpenStack & Docker – What's Possible Now



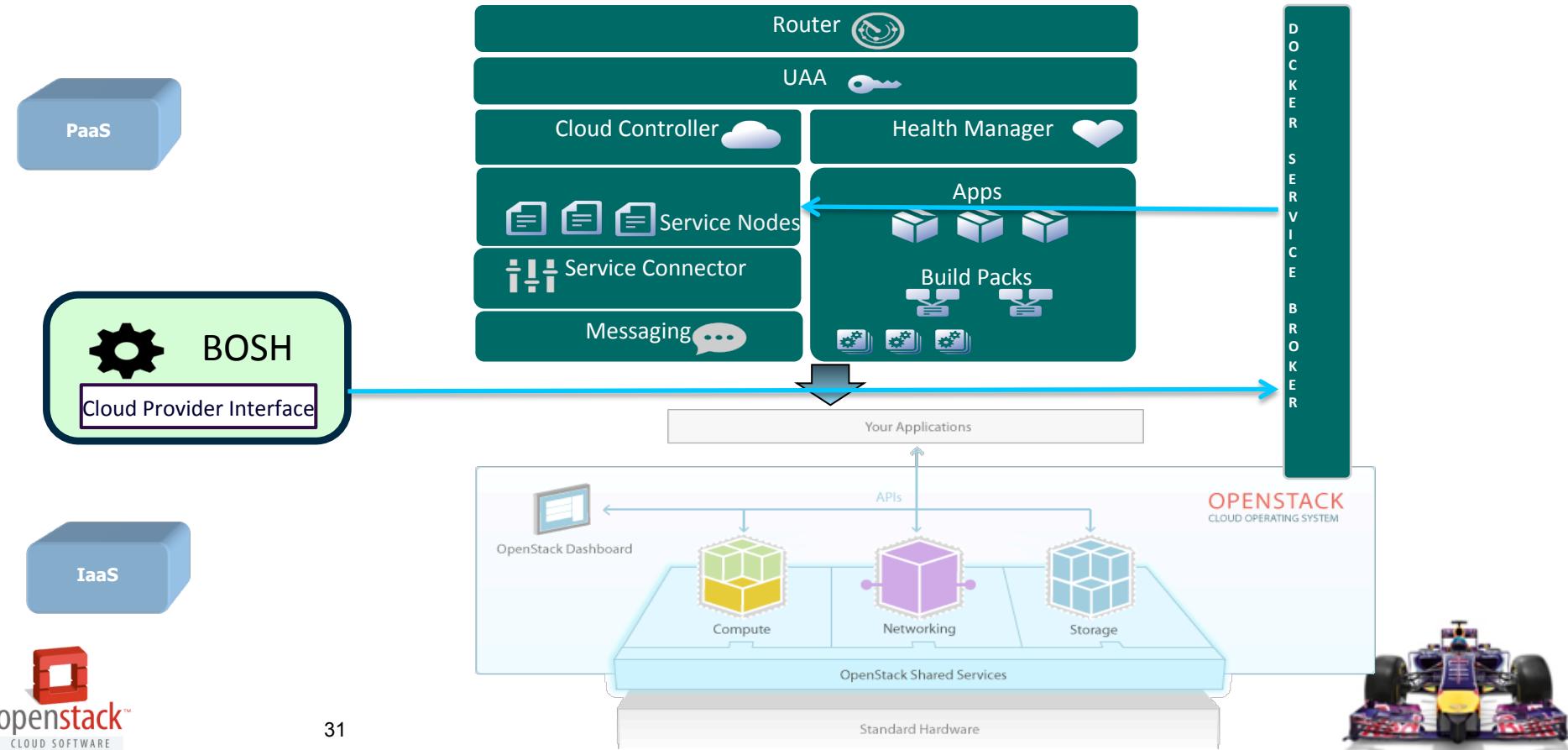
Cloud Foundry , OpenStack & Docker – What's Possible Now



Cloud Foundry , OpenStack & Docker – What`s Coming



Cloud Foundry , OpenStack & Docker – What`s Can be future



References and Links

- [Diego Design Notes](#)
- [CF Summit Keynote \(Slides\)](#)
- [Cloud Foundry: Diego Explained](#)
- [Docker OpenStack Wiki](#)
- [Heat for Docker](#)
- [More Heat for Docker](#)
- [Docker Service Broker Blog Post](#)



IBM Sponsored Sessions on Wednesday, November 5th

9:50 in Room 243

Step on the Gas: See how Open Technologies are driving the future of the enterprise

Todd Moore, Director, Open Technologies and Partnerships

David Lindquist, Chief Technology Officer / Vice President Strategy and Architecture

11:50 Room 212/213

IBM and OpenStack: Collaborations beyond the code

Manuel Silveyra, Daniel Krook

13:50 Room 212/213

A Use Case Driven view of IBM's OpenStack based Offerings

Moe Abdula, Vice President of Cloud Strategy

14:40 Room 212/213

IBM OpenStack Offerings in Action

Moe Abdula, Vice President of Cloud Strategy



IBM Sessions on Monday, November 3rd

15:20 R.251	When Disaster Strikes the Cloud: Who, What, When, Where and How to recover	Ronen Kat, Michael Factor, and Red Hat
11:40 A.Blue	IPv6 Features in OpenStack Juno	Xu Han Peng, Comcast, and Cisco
15:20 R252	Why is my Volume in 'ERROR' State!?! An Introduction to Troubleshooting Your Cinder Configuration	Jay Bryant
16:20 A.Blue	Group Based Policy Extension for Networking	Mohammad Banikazemi, Cisco, Midokura, and One Convergence

IBM Sessions on Tuesday, November 4th

11:15 R252	The perfect match: Apache Spark meets Swift.	Gil Vernik, Michael Factor, and Databricks
15:40 R242	Docker Meets Swift: A Broadcaster's Experience	Eran Rom, and RAI
16:40 Mailot	User Group Panel: India, Japan, China	Ying Chun Guo, Guang Ya Liu, Qiang Guo Tong
14:50 Passe	A Practical Approach to Dockerizing OpenStack High Availability	Manuel Silveyra, Shaun Murakami, Kalonji Bankole, Daniel Krook

IBM Sessions on Wednesday, November 5th

09:00 R241	Monasca DeepDive: Monitoring at scale	Tong Li , Rob Basham, HP and Rackspace
09:00 R242	Beyond 86: Managing multi-platform environments with OpenStack	Shaun Murakami, Philip Estes
09:50 R253	Troubleshooting Problems in Heat Deployments	Fabio Oliveira,Ton Ngo,Priya Nagpurkar, Winnie Tsang
11:50 R251	Keystone to Keystone Federation Enhancements for Hybrid Cloud Enablement	Steve Martinelli, Brad Topol, CERN, and Rackspace
17:50 R253	Practical advice on deployment and management of enterprise workloads	Jarek Miszczyk, Venkata Jagana



Accelerating Enterprise OpenStack - 0 to Kilo in four short years



Optimize
processes to save time and cost



Deploy
and manage hybrid IT

Reduce
provisioning time by up to 95%

Build
on workload optimized infrastructure

Learn more at these IBM sponsored sessions on Wednesday:

9:50 Room 243

Step on the Gas: See how Open Technologies are driving the future of the enterprise

11:50 Room 212/213

IBM and OpenStack: Collaborations beyond the code

1:50 Room 212/213

A Use Case Driven view of IBM's OpenStack based Offerings

2:40 Room 212/213

IBM OpenStack Offerings in Action

Stop by the IBM Booth (B4)
Demos, games and FREE tee shirt.





Legal Disclaimer

- © IBM Corporation 2011. All Rights Reserved.
- The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
- If the text contains performance statistics or references to benchmarks, insert the following language; otherwise delete:
Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- If the text includes any customer examples, please confirm we have prior written approval from such customer and insert the following language; otherwise delete:
All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.
- Please review text for proper trademark attribution of IBM products. At first use, each product name must be the full name and include appropriate trademark symbols (e.g., IBM Lotus® Sametime® Unyte™). Subsequent references can drop "IBM" but should include the proper branding (e.g., Lotus Sametime Gateway, or WebSphere Application Server). Please refer to <http://www.ibm.com/legal/copytrade.shtml> for guidance on which trademarks require the ® or ™ symbol. Do not use abbreviations for IBM product names in your presentation. All product names must be used as adjectives rather than nouns. Please list all of the trademarks that you use in your presentation as follows; delete any not included in your presentation. IBM, the IBM logo, Lotus, Lotus Notes, Notes, Domino, Quickr, Sametime, WebSphere, UC2, PartnerWorld and Lotusphere are trademarks of International Business Machines Corporation in the United States, other countries, or both. Unyte is a trademark of WebDialogs, Inc., in the United States, other countries, or both.
- If you reference Adobe® in the text, please mark the first use and include the following; otherwise delete:
Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
- If you reference Java™ in the text, please mark the first use and include the following; otherwise delete:
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- If you reference Microsoft® and/or Windows® in the text, please mark the first use and include the following, as applicable; otherwise delete:
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- If you reference Intel® and/or any of the following Intel products in the text, please mark the first use and include those that you use as follows; otherwise delete:
Intel, Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- If you reference UNIX® in the text, please mark the first use and include the following; otherwise delete:
UNIX is a registered trademark of The Open Group in the United States and other countries.
- If you reference Linux® in your presentation, please mark the first use and include the following; otherwise delete:
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Other company, product, or service names may be trademarks or service marks of others.
- If the text/graphics include screenshots, no actual IBM employee names may be used (even your own), if your screenshots include fictitious company names (e.g., Renovations, Zeta Bank, Acme) please update and insert the following; otherwise delete: All references to [insert fictitious company name] refer to a fictitious company and are used for illustration purposes only.

