Package once / Run anywhere

Big Data and HPC workloads

Tryggvi Larusson
CTO & Co-founder GreenQloud
What is GreenQloud?

Hybrid & Private IaaS Cloud
## Infrastructure Overview

### Zone 1

<table>
<thead>
<tr>
<th>Resource</th>
<th>Allocated</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>6.25 GHz / 62.40 GHz</td>
<td>10.02%</td>
</tr>
<tr>
<td>Memory</td>
<td>4.25 GB / 126.09 GB</td>
<td>3.37%</td>
</tr>
<tr>
<td>Storage</td>
<td>Primary: 1.93 TB / 12.00 TB</td>
<td>16.69%</td>
</tr>
<tr>
<td></td>
<td>Secondary: 1017.65 GB / 6.00 TB</td>
<td>16.17%</td>
</tr>
<tr>
<td>Addresses</td>
<td>Public IPs: 32 / 40</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>Management IPs: 3 / 10</td>
<td>36%</td>
</tr>
</tbody>
</table>

### Baremetal

<table>
<thead>
<tr>
<th>Resource</th>
<th>Allocated</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>0.00 MHz / 192.00 GHz</td>
<td>0%</td>
</tr>
<tr>
<td>Memory</td>
<td>0.00 bytes / 512.00 GB</td>
<td>0%</td>
</tr>
</tbody>
</table>
Problem:

How to conveniently package, distribute and deploy your application?
Common pain points today

Regarding any software deployment on **Cloud, HPC Clusters or local environment**

- Every environment is **slightly** different
- OS, Networking, Libraries, Compilers etc.
Solution:

Build all your application dependencies into a binary “docker” container image
Docker Basics

- Container runtime on Linux
- “Standard” container image format
- Centralized repository of images:
  - [https://hub.docker.com/](https://hub.docker.com/)
  - (like github except for runtimes)
CoreOS Basics

- Very minimal Linux distribution
- Containing basically the kernel and a few unix utilities
- Main tools:
  - systemctl for running services
  - etcd for clustering hosts together
  - fleet for distribution across cluster
  - docker for running applications
“Everything CoreOS is building has been inspired by the way Google runs its data center infrastructure”

- Alex Polvi, CEO CoreOS

Docker Advantages

- Simple Packaging
- Thousands of ready made images
- Low overhead (compared to VMs)
- Portability (Build once run anywhere)
- Composable
  - (One step closer to the Lego block dream)
Docker Build:
Dockerfile

FROM ubuntu:14.04
MAINTAINER Tryggvi Larusson <tryggvi@greenqloud.com>
RUN apt-get update
RUN apt-get -y install mysql-server
EXPOSE 3306
CMD ["/usr/bin/mysqld_safe"]

docker build -t="mysql" .
Docker Run:

docker run -d -p 3306:3306 mysql

- daemonize
- port mapping
- container name
Where is this leading us?
State of the Cloud today
## Development of Computing

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mainframes</td>
<td>Minicomputers</td>
<td>PCs</td>
<td>PCs</td>
<td>PCs + ASP</td>
<td>Cloud</td>
<td>Distributed - Federated Cloud</td>
</tr>
</tbody>
</table>

- **1960s:** Mainframes, Centralized, Closed Source
- **1970s:** Minicomputers, Centralized, Closed Source
- **1980s:** PCs, Decentralized, Closed Source
- **1990s:** PCs, Decentralized, Closed Source
- **2000s:** PCs + ASP, Hybrid, Closed + Open Source
- **2010s:** Cloud, Centralized, Closed + Open Source
- **2020s:** Distributed - Federated Cloud, Decentralized, Open Source
Development of the cloud

PHASE 1
Virtualization

PHASE 2
Cloud

PHASE 3
Hybrid Cloud

PHASE 4
Cloud Federation

Local Datacenters + Virtualization

Public Cloud

Private Cloud

Public Cloud

Hybrid Cloud

Private Cloud

Present

Cloud Federation
Future of the cloud is distributed

Standard application packaging + containers are the key
Clouds of tomorrow
Key Future Enabling Technologies for the Distributed Cloud
Demo
Container vs. VMs

Containers are isolated, but share OS and, where appropriate, bins/libraries.
Linpack

STREAM benchmark 16 Sockets

STREAM benchmark 1 Socket

Run anywhere

- Cloud Instance
- KVM VM
- VMWare VM
- Baremetal Machine
Implications for HPC developers/admins

- **Freedom of choice**
  - Have more options in selecting where your workloads actually run - easier to experiment

- **Speed of deployment**
  - Standard containers make it very easy to automate

- **Collaboration**
  - Easily share your HPC application to others and get feedback
  - Build on an HPC app/container somebody already built and improve it
Thank you!

Questions?

@tryggvil
tryggvi@greenqloud.com