S111017- Implementing DevOps and Hybrid Cloud

Srihari Angaluri
Lenovo Data Center Group

Red Hat Summit 2017
5/3/2017
Outline

• DevOps and Containers
• Architectural Considerations
• Lenovo Cloud Technology Center
• Implementing Red Hat OpenShift
• Hybrid Cloud Management with CloudForms
• Lenovo Converged Platform
• Demo
Most Companies Have:

- A Lack of effective and modern processes & principals
- Average to poor inter-department collaboration
- Tools that are not interconnected in the workflow
- Lots of software re-coding “rollbacks”
- Infrastructure challenges; usually can’t get what they need – when they need it
- Control & governance concerns
- Issues they don’t know how to fix (in some cases know they have issues)
Industry Evolution

Traditional
- Acquired as discrete parts
- Hard to deploy

Wave 1: Converged
- Acquired as single entity
- Faster deployment

Wave 2: Hyper Converged
- Moves storage into server

Wave 3: Software Defined Infrastructure (SDI)
- IT Agility - virtualizes server, storage, network, and public cloud resources

Applications / Workloads

Value

IT Agility
- Rapid time-to-deployment
- Ease of management
- Continuous and dynamic
Containers

Linux Containers (LXC)

Containers Vs VMs

https://www.docker.com/what-container
Run Your Workloads, Where Its Best For You

- Elastic Workloads
  - Spin up and down resources on the public cloud
  - Some are best in public cloud, some on Premise
  - Depends on need – amounts, time, costs

- Predictable Workloads
  - Lower costs with private cloud infrastructure

Your % Will Vary – You Choose

Balance **Owning** and **Renting** For Today’s Enterprise Workloads
Hybrid-Cloud – Integrating 3-Tier, Web-Scale, Public Cloud

- Maximize legacy infrastructure where cost-effective
- Enable Hybrid-Cloud infrastructure & cloud-native tools

- Leverage “Prem” and “Public” resources when needed for specific applications and workloads
Container Platform Architecture
Traditional, Stateful, and Microservices-based Apps

- Business Automation: Container
- Integration: Container
- Data & Storage: Container
- Web & Mobile: Container

Self-Service

Service Catalog
(Language Runtimes, Middleware, Databases)

Build Automation
Deployment Automation

OpenShift Application Lifecycle Management
(CI/CD)

Container Orchestration & Cluster Management
(kubernetes)

Networking
Storage
Registry
Logs & Metrics
Security

Infrastructure Automation & Cockpit

Enterprise Container Host

Container Runtime & Packaging
(Docker)

Atomic Host
Red Hat Enterprise Linux
Google Kubernetes Container Orchestrator

Kubernetes Features

- **Automatic binpacking**
  Automatically places containers based on their resource requirements and other constraints, while not sacrificing availability. Mix critical and best-effort workloads in order to drive up utilization and save even more resources.

- **Horizontal scaling**
  Scale your application up and down with a simple command, with a UI, or automatically based on CPU usage.

- **Automated rollouts and rollbacks**
  Kubernetes progressively rolls out changes to your application or its configuration, while monitoring application health to ensure it doesn’t kill all your instances at the same time. If something goes wrong, Kubernetes will rollback the change for you. Take advantage of a growing ecosystem of deployment solutions.

- **Self-healing**
  Restarts containers that fail, replaces and reschedules containers when nodes die, kills containers that don’t respond to your user-defined health check, and doesn’t advertise them to clients until they are ready to serve.

- **Service discovery and load balancing**
  No need to modify your application to use an unfamiliar service discovery mechanism. Kubernetes gives containers their own IP addresses and a single DNS name for a set of containers, and can load-balance across them.

- **Secret and configuration management**
  Deploy and update secrets and application configuration without rebuilding your image and without exposing secrets in your stack configuration.

- **Storage orchestration**
  Automatically mount the storage system of your choice, whether from local storage, a public cloud provider such as GCP or AWS, or a network storage system such as NFS, iSCSI, Gluster, Ceph, Cinder, or Flocker.

- **Batch execution**
  In addition to services, Kubernetes can manage your batch and CI workloads, replacing containers that fail, if desired.

https://kubernetes.io/
OpenShift Platform Architecture

https://docs.openshift.com/container-platform/3.5/architecture/index.html
Implementing OpenShift Container Platform
Lenovo Container Orchestration Platform

An integrated HW and SW reference architecture, purpose built for container applications. Easy to deploy, easy to scale, easy to manage.

- Integrated container application platform (PaaS) reference architecture with enterprise grade OpenShift/Kubernetes and xClarity infrastructure management
- Simplified deployment (minutes), easily scale across racks
- Highly optimized for scalability, performance and value (Intel SSD technology, Intel Xeon, Intel NIC’s)

Leaf/spine switch architecture to scale racks horizontally

Seamless scaling of compute or storage capacity as you grow

Lenovo x3550 M5
Server for compute building block

Lenovo x3650 M5
Server for SDS storage building block

Self-Service

Multi-Language

Automation

Collaboration

OPENSHIFT®
by Red Hat®

Standards Based

Web Scale

Open Source

Enterprise Grade

CPUs

SSDs

Network
Hardware Architecture and Considerations for OpenShift

A node provides the runtime environments for containers. Each node in a Kubernetes cluster has the required services to be managed by the master. Nodes also have the required services to run pods, including the Docker service, a kubelet, and a service proxy.

OpenShift Container Platform leverages the Kubernetes persistent volume (PV) framework to allow administrators to provision persistent storage for a cluster. Using persistent volume claims (PVCs), developers can request PV resources without having specific knowledge of the underlying storage infrastructure.
OpenShift Deployment Automation

- Baremetal initial discovery
- IMM/IPMI configuration
- Firmware updates
- Network configuration
- OS deployment
- Embedded DNS, DHCP, TFTP

Management node xCAT docker image

- Generates cluster HW manifest and topology (hosts, IP addresses, inventory, etc.)
- Tool to translate topology to Ansible OpenShift configuration file
- Kick-off ansible installation playbook
- Cluster is deployed and ready for operation
Create an OSEv3 group that contains the master, nodes, etcd, and lb groups. The lb group lets Ansible configure HAProxy as the load balancing solution. Comment lb out if your load balancer is pre-configured.

```yaml
[OSEv3:children]
masters
etcd
lb
```

Set variables common for all OSEv3 hosts

```yaml
[OSEv3:vars]
anible_ssh_user=root
deployment_type=openshift-enterprise
openshift_master_default_subdomain=apps.oshift.local
openshift_master_identity_providers=[{'name': 'htpasswd_auth', 'login': 'true', 'challenge': 'true', 'kind': 'HTPasswdPasswordIdentityProvider', 'filename': '/etc/origin/master/htpasswd'}]
openshift_master_cluster_method=native
openshift_master_cluster_hostname=haproxy-0
openshift_master_cluster_public_hostname=haproxy-0.oshift.local
openshift_hosted_metrics_public_url=master-0.oshift.local
```

Enable ntp on masters to ensure proper failover

```yaml
openshift_clock_enabled=true
```

Host group for masters

```yaml
[masters]
master-0
master-1
master-2
```

Host group for etcd

```yaml
[etcd]
master-0
master-1
master-2
```

Specify load balancer host

```yaml
[lb]
haproxy-0
```

Host group for nodes, includes region info

```yaml
[nodes]
master-0:2 openshift_node_labels="{'region': 'infra', 'zone': 'default'}"
app-0 openshift_node_labels="{'region': 'primary', 'zone': 'east'}"
app-1 openshift_node_labels="{'region': 'primary', 'zone': 'west'}"
app-2 openshift_node_labels="{'region': 'primary', 'zone': 'west'}"
```
Virtualized OpenShift Enterprise 3.5 Implementation

CloudForms Management Engine Appliance

OpenShift 3.5 “virtual” Cluster RHEL 7.3 Hosts
OpenShift Deployment Architecture

- CloudForms (multi cloud manager)
- xClarity Administrator (hardware manager)
- OpenShift Console (user self-service)
- User Applications (deployed in OpenShift)

External Cluster Services: DNS, NTP

CloudForms Installer (ansible host)

OpenShift Cluster

haproxy-0

master-0
app-0

master-1
app-1

master-2
app-2
```bash
[root@master-0 ~]# oc get nodes --show-labels
NAME                STATUS    AGE             LABELS
app-0.oshift.local  Ready     1d         beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
app-1.oshift.local  Ready     1d         beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
app-2.oshift.local  Ready     1d         beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
master-0.oshift.local Ready     1d          beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
master-1.oshift.local Ready     1d          beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
master-2.oshift.local Ready     1d          beta.kubernetes.io/arch=amd64,beta.kubernetes.io/
```

```bash
[root@master-0 ~]# oc get projects
NAME               DISPLAY NAME DOWNLOAD STATUS
cake-project       cake-phy-mysql Active
default            Active
kube-system        Active
logging            Active
management-infra  Active
nexus3-demo        nexus3-example Active
openShift          Active
openShift-infra    Active
```

```bash
[root@master-0 ~]# oc get pods --all-namespaces
NAMESPACE     NAME                  READY     STATUS      RESTARTS   AGE
deploy         db                    1/1       Running     0          1d
openshift      mysql                 1/1       Running     0          1d
```

```bash
[root@master-0 ~]# oc get routes --all-namespaces
NAMESPACE     NAME                  HOST/PORT      PORT      TERMINATION   WILDCARD
```
Creating and deploying your containerized apps
Built-in templates for common runtimes and usecases

Languages
- Java
- JavaScript
- .NET
- Perl
- PHP
- Python
- Ruby

Technologies
- Business Process Services
  Model, automate, and orchestrate business processes across applications, services, and data.
- Continuous Integration & Deployment
  Automate the build, test, and deployment of your application with each new code revision.
- Data Stores
  Store and manage collections of data.
Managing your Hybrid Cloud with CloudForms
Managing OpenShift with CloudForms
OpenShift metrics integration for chargeback

Hawkular Metrics
A time series metrics engine based on Cassandra
0.21.5.Final-redhat-1
Get SHA - 62030bf123c56a6a6aadb1e6f7114dca37edfb708
Metrics Service STARTED
Managing your infrastructure with Lenovo xClarity Administrator
XClarity™ Administrator

Version 1.2.2

Language: English US

* User name: admin

* Password: **********

Log In

Licensed Materials - Property of Lenovo.
© Copyright Lenovo 2016. All Rights Reserved.
## Alerts

Alerts indicate hardware or management conditions that need investigation and user action.

<table>
<thead>
<tr>
<th>Severity</th>
<th>Serviceability</th>
<th>Date and Time</th>
<th>System</th>
<th>Alert</th>
<th>System Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>Not Available</td>
<td>Apr 24, 2017, 2:49:05 PM</td>
<td>cmrn02</td>
<td>Uncorrectable error detected for memory device 14 in Group 1 on Subsystem System Memory.</td>
<td>Chassis</td>
</tr>
<tr>
<td>Critical</td>
<td>Not Available</td>
<td>Apr 24, 2017, 2:47:12 PM</td>
<td>cmrn02</td>
<td>Uncorrectable error detected for memory device 14 in Group 1 on Subsystem System Memory.</td>
<td>Chassis</td>
</tr>
<tr>
<td>Critical</td>
<td>Support</td>
<td>Apr 24, 2017, 2:45:13 PM</td>
<td>cmrn02</td>
<td>An Uncorrectable Error has occurred on CPUs.</td>
<td>Chassis</td>
</tr>
<tr>
<td>Warning</td>
<td>Support</td>
<td>Apr 24, 2017, 2:45:26 PM</td>
<td>cmrn02</td>
<td>The PFA Threshold limit (correctable error logging limit) has been exceeded on DIMM number Low at address 2. MCO S</td>
<td>Chassis</td>
</tr>
<tr>
<td>Critical</td>
<td>Not Available</td>
<td>Apr 24, 2017, 2:44:01 PM</td>
<td>cmrn02</td>
<td>An Uncorrectable Error has occurred on CPUs.</td>
<td>Chassis</td>
</tr>
</tbody>
</table>

## Logs

The Event log provides a history of hardware and management conditions that have been detected.

<table>
<thead>
<tr>
<th>Severity</th>
<th>Serviceability</th>
<th>Date and Time</th>
<th>System</th>
<th>Event</th>
<th>System Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:57:54 PM</td>
<td>Management Server</td>
<td>The Service Data archive for event 000F08132594FF01 from c102x01-1mm has been collected successfully.</td>
<td>Management</td>
</tr>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:47:03 PM</td>
<td>Management Server</td>
<td>The Service Data archive for event 000F08132594FF01 from c102x01-1mm has been collected successfully.</td>
<td>Management</td>
</tr>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:44:47 PM</td>
<td>Management Server</td>
<td>The management server started collecting the Service Data archive for event 000F08132594FF01 generated on CPUs in the 102x01-1mm.</td>
<td>Management</td>
</tr>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:44:56 PM</td>
<td>Management Server</td>
<td>Event 000F08132594FF01 generated on c102x01-1mm has been added to the Service Data collecting process.</td>
<td>Management</td>
</tr>
<tr>
<td>Critical</td>
<td>Support</td>
<td>Apr 24, 2017, 2:44:16 PM</td>
<td>cmrn02</td>
<td>An Uncorrectable Error has occurred on CPUs</td>
<td>Chassis</td>
</tr>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:41:02 PM</td>
<td>Management Server</td>
<td>The management server started collecting the Service Data archive for event 000F08132594FF01 generated on c102x01-1mm.</td>
<td>Management</td>
</tr>
<tr>
<td>Informal</td>
<td>Not Required</td>
<td>Apr 24, 2017, 2:41:11 PM</td>
<td>Management Server</td>
<td>Event 000F08132594FF01 generated on c102x01-1mm has been added to the Service Data collecting process.</td>
<td>Management</td>
</tr>
<tr>
<td>Warning</td>
<td>Support</td>
<td>Apr 24, 2017, 2:41:01 PM</td>
<td>cmrn02</td>
<td>The PFA Threshold limit (correctable error logging limit) has been exceeded on DIMM number Low at address 2. MCO S</td>
<td>Chassis</td>
</tr>
</tbody>
</table>
More information

• Contact:
  – Srihari Angaluri sangaluri@lenovo.com
    Solution Architect
    Lenovo Data Center Group

• https://www.redhat.com/en/technologies/cloud-computing/openshift
• https://www.redhat.com/en/containers/what-is-kubernetes