z Systems and Cloud Computing

Understanding the Direction

Paul Hall
phall@ca.ibm.com
What is Cloud?

- Cloud is not a product
- Cloud is not a technology
- Cloud is a methodology for delivering computing services
- Cloud is the ability for users to provision and expose new environments or services:
  - In a self-service fashion
  - Backed by automation
  - In a manner that preserves the security and integrity of the environment.
Why are customers interested in Cloud?

- In a word: Agility

- Businesses cannot afford to wait for computer infrastructure at traditional speeds

- The complexity and scale required to support today’s requirements may not be pragmatically achievable or economically viable with traditional technologies/methods.
Cloud Computing - Based on Virtualization and Standardization

We need to understand that Cloud computing is a journey beginning with virtualization and consolidation of environments and ending with workload pattern-based deployment of IT services.

Cloud Computing – Characteristics*:
- Rapid elasticity
- Broad network access
- Resource pooling
- Measured service
- On-demand self-service

* Source: National Institute of Standards and Technology (NIST)
Cloud deployment models

- All in-house IT
  - Applications
  - Data
  - Runtime
  - Middleware
  - O/S
  - Virtualization
  - Servers
  - Storage
  - Networking

- Infrastructure As A Service (IaaS)
  - Applications
  - Data
  - Runtime
  - Middleware
  - O/S
  - Virtualization
  - Servers
  - Storage
  - Networking

- Platform as a Service (PaaS)
  - Applications
  - Data
  - Runtime
  - Middleware
  - O/S
  - Virtualization
  - Servers
  - Storage
  - Networking

- Software as a Service (SaaS)
  - Applications
  - Data
  - Runtime
  - Middleware
  - O/S
  - Virtualization
  - Servers
  - Storage
  - Networking
IBM Deployment models*

- “Private” means built, run and managed by internal IT organization
- “Public” means built and supported by an external organization
- “The “on-prem/off-prem” axis determines where the systems are physically located

*only IBM cloud services are listed on this chart – but there are many other offerings from a variety of companies
z/OS
IBM Vision and Strategy

- Transform z/OS customers into service providers
  - Provide world class services internally to their consumers
  - Provide services on the internet
  - Purchase services from others

- Assist z/OS customers in seeing IT as a Value vs. Cost
  - Move from cost savings to value generation
  - Include metrics, capping, multi-tenancy

- Improve configuration and deployment of z/OS Software
  - Improve time to value
  - Creating an advantage for z/OS
  - Include workflows, resource pooling, rapid provisioning, etc…
Implementation Scope

- Provide necessary tooling and infrastructure to allow customers to expose z/OS and z/OS services in a Private Cloud environment
  - We plan to provide all the necessary tools to allow a system programmer to create a z/OS on-premise private cloud environment without the need of any IBM Services.

- Workloads composed on multi-platform implementations, this will allow z/OS to participate in a Hybrid Cloud environment
  - This participation continues the practice of limiting access to z/OS to a customer’s intranet, but allowing the function to be part of a larger Cloud service

- Furthermore, the Cloud workloads accessing z/OS will allow for multiple tenants to reside on a single z/OS host.
  - Multitenancy is a typical approach in lowering costs by sharing platform resources
Focus of z/OS Cloud

External ecosystem

Marketplace

Services Catalog

Services

Caching
Key Serving
Cryptography
Payment
services

API Economy

SaaS

IOC
SAP

Cloud Operating Environment

PaaS

Traditional middleware-based workloads

Services & composition patterns

API & integration services

CICS
DB2
IMS
WebSphere AS
MQ

Resource abstraction & optimization

Software defined compute

Software defined storage

Software defined networking

Hardware

Private Clouds

Hybrid

Public Clouds
Cloud Computing on z/OS

With z/OS, we need to think about cloud just a bit differently…..

- Today in cloud environments on distributed servers, or even with Linux on z Systems, customers would provision a virtual machine with an instance of an operating system to run a single workload.
  - To deploy another workload would mean another virtual machine with another instance of the operating system.

- However, in the context of z/OS, this methodology goes against everything we have come to know and expect about z/OS.
  - On z/OS, you have the ability to run multiple disparate workloads with different service levels for those hosted workloads with isolation or multitenancy.

- Hence our approach for cloud on z/OS is not focusing on the provisioning of operating system instances, but rather the ability to provision multiple workloads in a single z/OS instance.
  - We make the assumption that the z/OS operating system is already part of the infrastructure and does not need to be instantiated to deploy a workload in a production environment.
  - Provisioning a new instance of a the z/OS operating system in a new LPAR is an important dev/test requirement, but out of scope of this work.
Architectural Overview

Cloud Services / Applications - business services

Workflow Users
CICS/IMS/MQ/DB2/WAS subsystems, Cloud Service Providers

UI / REST API
Workflow Engine
System Definitions, Registry
z/OSMF

z/OS Operating System

Resource Mapping Services
Tenant Mgmt
Resource Mgmt
WLM
RACF
SMF
TCP/IP
DFSMS

Email
SMF data

z/OSMF
Implementation View

- Bluemix
- ISV written Consumer Portal
- IBM written Service Manager

- Chef Scripts
- OpenStack HEAT Template
- Urban Code Deploy w/ Patterns

Dotted lines represent remote call boundaries

- z/OSMF
  - UI / REST API
  - Workflow Engine
  - System Definitions, Registry

- z/OSMF
  - Resource Mapping Services
  - Tenant Mgmt
  - Resource Mgmt
  - WLM
  - RACF
  - SMF

- z/OS Operating System
- Email
- SMF data

- Bluemix
- ISV written Consumer Portal
- IBM written Service Manager

- Chef Scripts
- OpenStack HEAT Template
- Urban Code Deploy w/ Patterns

Dotted lines represent remote call boundaries

- z/OSMF
  - UI / REST API
  - Workflow Engine
  - System Definitions, Registry

- z/OSMF
  - Resource Mapping Services
  - Tenant Mgmt
  - Resource Mgmt
  - WLM
  - RACF
  - SMF

- z/OS Operating System
- Email
- SMF data
A Shift for the Traditional System Programmer

- System programmers today are in the main line path of allocating system resources for new workload deployment.
  - With this new z/OS Cloud capability, policies can be pre-defined and resource pools pre-established that will still allow traditional z/OS system programmer to maintain control of system resources, but no longer require their presence during the allocation process.
  - The ability for customers to change when the decision point is made is a “temporal shift” that no longer requires the system programmer to be in the mainline path, but still provides all the necessary control with added flexibility.
  - Even with this automated provisioning capability, the system programmer can be responsible for “hardening” the changes.
  - Integration with customer’s change control mechanism and decision to activate instantly or deferred to a later time will be supported.
Summary of Cloud Characteristics - z/OS assessment

- **On-demand Self-Service**
  - z/OS the operating system could supply a lot more here
  - This is an area we need to work on

- **Broad Network Access**
  - TCP/IP Based – z/OS has this covered
  - Browser Managed – Working on this with z/OSMF

- **Resource Pooling**
  - z/OS is a shared everything configuration – a fundamental difference from typical cloud environments and likely a differentiating value – we need to demonstrate this
    - Critical resources are mediated by WLM and other parts of z/OS
  - z/OS needs to expose resource pooling API’s

- **Rapid Elasticity or Expansion**
  - z/OS can scale from <1GB and fractional “MIPS” to 4TB and 112,000 “MIPS”, then multiplied by up to 32 coupled systems
  - And then we can scale out to multiple Sysplexes

- **Measured Service**
  - z/OS captures a lot of data for this purpose in SMF records and other API’s
  - z/OS needs to pick up the ability to meter and cap usage or resources
We want our customers to think of themselves as Service Providers

We want the end-user experience to be as much self-service as possible. For example:

• Provision a DB2 table or an MQ queue
• Click on a tile in Bluemix for Middleware as a Service, such as a CICS region or a WAS server
• Provision a new user ID
• Stand up a new copy of an entire IMS subsystem
• At the click of a button, connect a new WAS server to an existing DB2 instance

We want IT organizations to capture value through charge back

Driving towards the functions of: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service

We want to remove barriers
Linux for z Systems
&
LinuxOne
Cloud Management for z Systems

- z Systems and LinuxOne are providing an open and standards-based approach to cloud management
  - OpenStack enablement of z/VM and KVM for IBM z

- z Systems and LinuxOne support and embrace any of the major industry ecosystem initiatives around software-defined infrastructure management, container technologies and Platform-as-a-Service.

- VMware
- Cloud Foundry
- Apache Mesos
- Kubernetes
- OpenShift
- Docker
Cloud Stack Architecture Leveraging Open Source

<table>
<thead>
<tr>
<th>Platform as-a-Service</th>
<th>Deployment Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenShift</td>
<td>Chef</td>
</tr>
<tr>
<td>Red Hat</td>
<td>Ansible</td>
</tr>
<tr>
<td>Cloud Foundry</td>
<td>Puppet</td>
</tr>
<tr>
<td>SUSE</td>
<td>Salt Stack</td>
</tr>
<tr>
<td>BlueMix</td>
<td>Juju</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Infrastructure as-a-Service</th>
<th>Container Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack</td>
<td>Docker</td>
</tr>
<tr>
<td>Nova</td>
<td>Kubernetes</td>
</tr>
<tr>
<td>Neutron</td>
<td>Mesos</td>
</tr>
<tr>
<td>Cinder</td>
<td>LXC</td>
</tr>
<tr>
<td></td>
<td>LXD</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Virtual Infrastructure</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SLES</td>
<td>Ubuntu</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>RHEL</td>
</tr>
<tr>
<td>z/VM</td>
<td>KVM for IBM z</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Physical Infrastructure</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM LinuxONE Rockhopper™</td>
<td></td>
</tr>
<tr>
<td>IBM LinuxONE Emperor™</td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td></td>
</tr>
<tr>
<td>Switches</td>
<td></td>
</tr>
</tbody>
</table>

Each Distro (SUSE, RedHat, Ubuntu) will have its own flavor of a cloud stack
Product Options for a Private Cloud on z Systems

- IBM Cloud Manager Appliance for z/VM (CMA)
  - OpenStack based provisioning of Linux images and topologies; manages from z/VM

- IBM Cloud Manager with OpenStack (ICM)
  - Same function as Cloud Manager Appliance; manages from x or p platforms. Version 4.3 adds support for KVM on z Systems

- IBM Cloud Orchestrator
  - Same function as CMA and ICM, with Orchestration, advanced Self-Service catalog, and external product integration functions

- VMware vRealize Automation
  - Provision Linux guests on z Systems from the vRealize cloud management platform

- SUSE OpenStack Cloud 6 (Beta announced 10/27/2015)
  - OpenStack private cloud software adds z/VM and KVM on z Systems support
KVM for IBM z Systems - A new hypervisor choice

The Kernel-based Virtual Machine (KVM) offering for IBM z Systems™ is software that can be installed on z Systems processors and can host Linux® on z Systems guest virtual machines.

- The KVM offering can co-exist with z/VM virtualization environments, z/OS®, Linux on z Systems, z/VSE® and z/TPF.
- Simplifies configuration and operation of server virtualization.
- The KVM offering is optimized for z Systems architecture and provides standard Linux and KVM interfaces for operational control of the environment, as well as supporting OpenStack® interfaces for virtualization management.
  - Enterprises can easily integrate Linux servers into their existing infrastructure and cloud offerings.
- Allows customers to leverage common Linux administration skills to administer virtualization.
- Provides an Open Source virtualization choice.
IBM and VMware announced a cooperative effort to give our mutual clients the ability to provision and manage virtual machines and applications running on IBM z Systems (and IBM Power Systems) with VMware's vRealize™ Automation™ 6.2 (vRA) solution through OpenStack enabled APIs.
Using VMware’s vRealize Automation (vRA), clients can provision and orchestrate virtualized workloads on z/VM and KVM for IBM z Systems through the OpenStack interfaces.

- Single cloud management tool across multiple environments in the enterprise cloud, including public cloud.
  - Single pane of glass
- vRA supports Infrastructure as a Service (IaaS) by passing workload management requests via OpenStack API’s to IBM z/VM and KVM on IBM z.
A private cloud on LinuxOne (or z13) is very efficient

Performance comparison based on IBM Internal tests comparing IBM z13 cloud with one comparably configured private x86 cloud and one comparably configured public cloud running an aggregation of light, medium and heavy workloads designed to replicate typical IBM customer workload usage in the marketplace. System configurations are based on equivalence ratios derived from IBM internal studies and are as follows: Public Cloud configuration: total of 219 instances (128 for light workloads, 64 for medium workloads and 27 for heavy workloads); x86 Cloud configuration: total of eleven x86 systems each with 24 Intel E7-8857 v2 3.0GHz cores, 512GB memory, and 7x400GB SSDs; z13 Cloud configuration: total of 32 IFLs, 3806GB memory, and Storwize v7000 with 47x400GB SSDs. Price comparison estimates based on a 3YR Total Cost of Ownership (TCO) using publicly available U.S. prices (including a 20% discount for middleware) current as of January 1, 2015. Public Cloud TCO estimate includes costs (US East Region) of infrastructure (instances, data out, storage, support, free tier/reserved tier discounts), middleware and labor. z13 and x86 TCO estimates include costs of infrastructure (system, memory, storage, virtualization, OS, cloud management, middleware, power, floor space and labor). Results may vary based on actual workloads, system configurations, customer applications, queries and other variables in a production environment and may produce different results. Users of this document should verify the applicable data for their specific environment.
Thank You!