Software Defined Environments based on OpenStack and TOSCA
Disclaimer

References in content to IBM products, software, programs, services or associated technologies do not imply that they will be available in all countries in which IBM operates. Content, including any plans contained in content, may change at any time at IBM's sole discretion, based on market opportunities or other factors, and is not intended to be a commitment to future content, including product or feature availability, in any way. Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.

Please refer to the developerWorks terms of use for more information.
Agenda

• Introduction
• OpenStack and TOSCA
• SmartCloud Orchestrator as a first implementation of a „TOSCA Container“ based on OpenStack
• Software Defined Environments
• Summary
Mobility, big data, analytics, social collaboration and cloud are creating a new wave of business opportunities and IT challenges.

1. **Technology factors**
2. People skills
3. Market factors
4. Macro-economic factors
5. Regulatory concerns
6. Globalization
7. Socio-economic factors
8. Environmental issues
9. Geopolitical factors

**IBM Global CEO Study**

- **Speed Value**: 90% view cloud as critical to their plans
- **Extended Reach**: 1 Billion Smartphones and 1.2 billion mobile employees by 2014
- **Responsiveness**: 20B+ Intelligent business assets
- **New Insights**: 2.7ZB of digital content in 2012, up 50% from 2011
New models of product & service innovation

Systems of Record

- Data & Transactions
- App Infrastructure
- Virtualized Resources

New Modes of Engagement

- Expanding Interface Modalities
- Big Data and Analytics
- Social Networking

Next Generation Architectures

Data & Transaction Integrity

Smarter Devices & Assets

© 2012 IBM Corporation
Different levels of orchestration

Heterogeneous hybrid environments...

Resource Orchestration
Onboard, provision, manage

Workload Orchestration
Dynamic optimization

Service Orchestration
Lifecycle of cloud services
A layered and open cloud architecture is emerging

- **Business Applications as components**
  - Service Oriented Architecture

- **Platform Services**
  - PaaS
  - flexibility & discipline in patterns

- **Infrastructure Services**
  - IaaS
  - flexibility & discipline in infrastructure

- **Backplane**
  - Fit for purpose

- **Composition Layer**

- **Software Defined Environments**

- **Hardware**
Agenda

- Introduction
- **OpenStack and TOSCA**
- SmartCloud Orchestrator as a first implementation of a „TOSCA Container“ based on OpenStack
- Software Defined Environments
- Summary
OpenStack in a nutshell

Open source software for building private and public clouds.

Software

OpenStack Software delivers a massively scalable cloud operating system.

Working software that is constantly developed and enhanced

Community

A vivid community of developers and cloud computing experts, driven by different companies.

Cisco WebEx Runs OpenStack

An increasing number of enterprises either base their cloud implementations on OpenStack – or build on top of it!

screenshot from openstack website
Openstack projects – conceptual architecture

- **nova** Compute
  - Provides sample UI
  - Reference implementation of API usage
  - Use API for volumes for instances
  - Use API for network connectivity for instances
  - Use authentication service via API

- **swift** Object Store
  - Use API to store image files

- **glance** Image Library
  - Use API to manage images

- **cinder** Block Storage
  - Use API for volumes for instances

- **quantum** Network

- **keystone** Identity

- **horizon** Dashboard
TOSCA Standard enables
- Portability and Interoperability of Cloud Services
- Model Driven Cloud Service Management
- "Appstore" for Cloud Services
- Open Hybrid Clouds
TOSCA – Technical Overview

Topology and Orchestration Specification for Cloud Applications

A language for defining Service Templates …

… including a Topology Template describing the structure of a service

… including the definition of building blocks for services

… including the definition implementation artifacts for manageability operations

… including the definition plans for orchestrating the application

TOSCA defines a packaging format (CSAR) for packaging models and all related artifacts.
Agenda

• Introduction
• OpenStack and TOSCA
• SmartCloud Orchestrator as a first implementation of a „TOSCA Container“ based on OpenStack
• Software Defined Environments
• Summary
Orchestration of Cloud Services based on a Common Cloud Stack

Leveraging Topology and Orchestration Specification for Cloud Applications (TOSCA)

Workload-driven Patterns

With Flexible and extensible deployment choices

SmartCloud Orchestrator
- SmartCloud Provisioning: Automate Optimized Workloads
- SmartCloud Entry: Automate IT Delivery
- Customer integrated hardware

SmartCloud Provisioning
- Automate Optimized Workloads

SmartCloud Entry
- Automate IT Delivery

PureFlex System

PureApplication System

Exploiting an open infrastructure base

Hybrid Cloud Interaction

© 2012 IBM Corporation
High level architecture
SmartCloud Orchestration and Provisioning

IBM Tivoli Monitoring (ITM)
SmartCloud Cost Management (SCCM)

SCOrchestrator
- Automation Modeling UI
  - IBM Business Process Manager (BPM)

Self Service
- Offering Catalog

Modeling and Admin
ICCT Image Creation
Image Library

Automation Engine
BPM Process Server

Composite Patterns Management
Hypervisor Management
OpenStack Gateway

Hybrid Extension
OpenStack

Content packages
- Amazon EC2
- VMWare vCenter
- KVM
- Openstack Cinder drivers for Block Storage

- Service Desk
- Licence Mgmt
- Accounting
- Cost Mgmt
- Network FW/LB
- Storage File-based
- SmartCloud Enterprise
- VMControl Power
- HyperV
- ESX
- XEN

*) supported in following releases
A typical scenario: create a new cloud service to deploy and manage SAP

Step 1: Cloud Admin: Import or define the structural model of the Cloud Service
Step 1 cont.: Cloud Admin: Import or define the process model of the Cloud Service

- **Palette of library assets enable easy workflow composition through drag and drop**
- **Access to rich libraries (toolkits) of reusable automation assets that enable to speed automation creation**
- **Graphical editor for composing and connecting workflows**
- **Tooling to edit, version, debug, optimize workflows**
- **Actions types, flow control, data handling primitives that simplify creation of complex automations**
- **Easy workflow action editing for managing: data mapping, error recovery options, implementation details, etc.**
Step 2: Cloud Admin: Publish service in the catalogue
Step 3 – End User: Request the service –
Fully automated, standardized, with a simple and intuitive interface
Agenda

• Introduction
• OpenStack and TOSCA
• SmartCloud Orchestrator as a first implementation of a „TOSCA Container“ based on OpenStack
• Software Defined Environments
• Summary
What is a Software Defined Environment (SDE)?

A new approach to IT service delivery, utilizing a programmable open standards-based foundation as an enabler for cloud, mobile and other dynamic enterprise solutions.

**SDE Characteristics**

- **Open industry API integration** encourages broad ecosystem of solutions providers.
- Workloads **dynamically** assigned to resources based on app characteristics and best available resources.
- **Analytics-based** compliance checking reduces security exposure and business risk.
- **Continuous optimization** to instantly address infrastructure issues and improve response to business needs.
- **Proactive** management of IT resources to improve efficiency and control costs of service delivery.

---

**Simple, Responsive, Adaptive**
Software Defined Environment Use Case Scenario: Delivery of Hadoop Service

Business Opportunity: New “Ad-fraud detection” application that uses real-time correlation of transaction data with ad click log data

Service Development and Delivery: Using Hadoop service for correlation and log analytics

Deployment Configurations: Based on Cost, Performance, Security and Availability Requirements
Orchestration of Software Defined Environments

- Capture the software and infrastructure definition of workloads
- Link software patterns to infrastructure patterns based on requirements
- Automatically orchestrate deployment and update of workloads on SDI
- Enable and differentiate orchestration with analytics

Value: enables rapid and continuous delivery of diverse set of workloads with agility and optimization on programmable heterogeneous infrastructure leveraging reusable building blocks
Composable Patterns supporting different roles in the Eco System

- Flexible selection of deployment topologies
- Separation of concerns
- Integration of models delivered by different providers

**Supported use cases from user perspective:**
- Import self-contained models from application down to infrastructure
- Deploy fully refined models
- Import componentized models with separation of application and middleware/infrastructure
- Deploy applications with variable selection of infrastructure templates based on policies
- Deploy middleware/infrastructure only patterns
- Edit or create new application or middleware/infrastructure models based on known Node- and Relationship Types
Example: Orchestrating an Infrastructure Pattern

- Flexible composition of patterns (re)using **standardized** building blocks
- Allows an **ecosystem** of content providers and content **reuse**
- “Deployment workflows” dynamically created based on pattern and policies

CreateNode [ name=Compute_A, type=power; OS=aixOS15; cores=2, Memory=20GB ]

CreateNode [ name=Compute_B, type=x64; OS=linux15; core=20, Memory=2GB ]

CreateNetwork [ name=NET_1, type=LOW_LATENCY; security=L2, Capacity=40 ]

AddMembers [ network=NET_1; node=Compute_A.NIC1, Compute_B.NIC1 ]

CreateStorage [ name=STORAGE_A, type=HIGH-IOPS; size=500GB ]

AttachDisk [ computeNode=Compute_A, disk=STORAGE_A ]
Software Defined Environments and OpenStack Heat

- Cloud Applications
- Software Defined Environment
  - Workload Definition & Orchestration
    - Workload Definition
    - Orchestration, Optimization and Analytics
  - Software Defined Infrastructure
    - Software Defined Compute
    - Software Defined Network
    - Software Defined Storage
  - Virtualized Network
  - Virtual Storage Layer
  - Heterogeneous Compute Resources

OpenStack Heat

Software Pattern

Infrastructure Pattern

- Compute
- Networking
- Storage

OpenStack Shared Services

Standard Hardware
Example for Modelling in SDE

SugarCRM two-tier deployment with scalable web tier

Single, self-contained model
Including scalable components

Building blocks for SugarCRM (TOSCA Node Types)

- SugarCRM Application
- SugarCRM Database
- Apache Webserver
- PHP Module
- MySQL
- Virtual Machine
- Load Balancer
Agenda

- Introduction
- OpenStack and TOSCA
- SmartCloud Orchestration as a first implementation of a “TOSCA Container” based on OpenStack
- Software Defined Environments
- Summary
Some Research Questions for SDE orchestration

• Generic Modelling Questions
  • Declarative vs. Imperative – when to use what? Define and Describe best practices
• Definition of the Base Model for SDS, SDN and SDC
  • What is the right granularity?
  • How do we link Software Patterns to Infrastructure Patterns?
  • Can we use more than one pattern engine and connect them via Reqs and Caps? If so, how do those engines interact?
• How do we manage SLAs and NFRs in SDE
  • How do we model and implement the autonomic behaviour of the SDE beyond deployment?
  • Implications on the TOSCA standard? For example: Need for standardization of eventing, signalling?
  • Imperative vs. Declarative approach wrt. NFRs and Policies?
  • „Autonomic Managers“ on various levels – how do they interact?
  • Where do we put optimization in the stack?
Summary

• Mobility, big data, analytics, social collaboration and cloud are creating a new wave of business opportunities and IT challenges
• IBM's open cloud architecture is based on emerging standards like OpenStack, TOSCA
• The Software Defined Environment (SDE) is composed of Software Defined Compute (SDC), Software Defined Storage (SDS), Software Defined Network (SDN) and an Orchestration component which allows to fully programatically compose, deploy, and manage all the elements which constitute the individual IT services.
• Resource and Workload Orchestration in SDE enables rapid and continuous delivery of diverse set of workloads leveraging reusable building blocks
• OpenStack Heat is an evolving orchestration engine for Software Defined Environments
• A new language called HOT based on the principles of TOSCA is currently being created for Heat
Backup
IT leaders are leveraging the transformational power of cloud to balance optimization of existing systems and innovation.

Optimization

Innovation

Fuels investments in innovation

Drives need for continuous IT optimization