State and the Art and Competitiveness Analysis

Working group coordinators
- Patrick Moreau – INRIA
- Lionel Seinturier – INRIA

Working group members
- François Exertier – Bull
- Denis Caromel – INRIA
- Cédric Carbone – Talend for Domain 3
- Jean-Pierre Dion – Bull for Domain 1
- Jean-Pierre Laisné – Bull for Domain 4
- Alexandre Lefebvre – Orange
- Jean-Pierre Lorré - Petals Link for Domain 2
- Hailong Sun – Beihang University
- Wang Wei - ISCAS

Document history

v1.0	PM	Nov 6, 2010	Initial version
v1.1	LS	Nov 9, 2010	Contribution on the initial version for the glossary and domains 1, 2 & 3
v1.2	PM	Nov 10, 2010	Format, some clarification on section 3
V1.3	PM	Dec 20, 2010: Integration of changes following the Dec 20, 2010 WG conf call.	
V1.4	FX	Jan 26, 2011: Contribution on the comparison matrix for JonAS + Jasmine + CLIF	
	DC	Jan 30, 2011: Contribution on the comparison matrix for ProActive	
	JPLo	Feb 02, 2011: Contribution on the comparison matrix for Petals	
V1.5	LS	Feb 25, 2011: Editing	
	JPD	Feb 25, 2011: Contrib. on the comparison matrix for OpenStack and OpenNebula	

1. Introduction

The objective of this document is to provide a state-of-the-art of the available OW2 solutions that contribute to the definition of an open source stack for cloud computing. The document is thus centered on existing OW2 projects that address concerns and challenges relating to cloud computing.

Projects are reviewed per domain (as identified by OSCi):

- Domain 1: Self-sizing and green PaaS
- Domain 2: Massively distributed services
- Domain 3: BI4Cloud
- Domain 4: Massively distributed cloud

This document provides some inputs for the consolidating objectives of the OSCi working group that are:

1. To establish the motivation to adopt Open source solutions for Cloud
2. To perform a macroscopic comparison of OSCi component set with state-of-the-art cloud stacks or component set
3. To perform a brick by brick comparison of OSCi components with state-of-the-art open source cloud components
4. To issue recommendations for an innovative roadmap

We start by proposing a glossary of common cloud computing terms and notions (Section 2). We then review domain per domain the existing OW2 projects that provide some solutions for building an open source cloud stack (Section 3).

2. Glossary

Several definitions for cloud computing have been proposed. Companies or academic institutions such as UC Berkeley, Burton Group, Gartner, McKinsey, Deloitte have proposed their own definition. Yet, it appears that the definition by the US National Institute of Standards and Technology (NIST)[1] is emerging as the leading and most referred to definition.

> *Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.*

Cloud computing is usually decomposed into three service models: SaaS, PaaS and IaaS. The following definitions are also provided by NIST.

> Software as a Service. *The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.*
>
> Platform as a Service. *The capability provided to the consumer is to deploy onto the cloud*

1 http://csrc.nist.gov/groups/SNS/cloud-computing/

infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

Infrastructure as a Service. *The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).*

Finally NIST has identified four deployment methods.

Private clouds. *The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.*

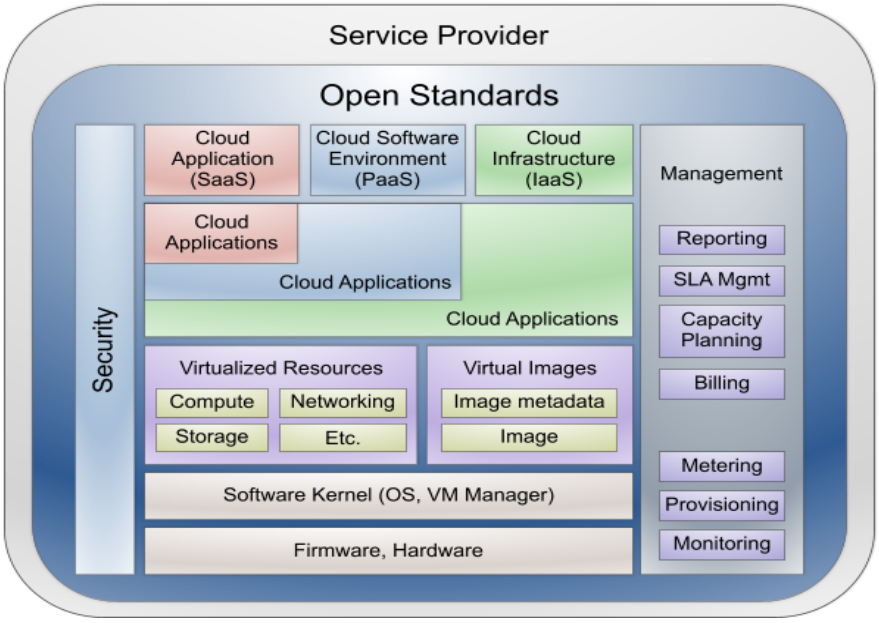
Community clouds. *The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.*

Public clouds. *The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.*

Hybrid clouds. *The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).*

Taxonomies for cloud features has emerged such as the one illustrated below that has been defined by the Open Cloud Manifesto[2]. The bottom layers are concerned with the hardware infrastructure and the management of virtualized resources. Two crosscutting activities are concerned with all levels of the stack: security and management. Among the management functions, the billing and service level agreement services play a central role in economical terms.

2 http://opencloudmanifesto.org



While cloud computing appears as a new solution to foster the development of IT, several existing technologies can be related to this domain: Grid computing, utility computing, virtualization, autonomic computing, service-oriented computing and architectures. These technologies matured in the past few years and have paved the way for the emergence of cloud technologies.

In terms of standardization, a lot of efforts is also devoted to cloud computing. We can mention the existence of the Cloud Standard Coordination that coordinates the activities of 13 standardization bodies with activities in the domain of cloud computing: OMG, DMTF, OGF, SNIA, OCC, CSA, ETSI, NIST, OG, ARTS, TM. Among these bodies, the DMTF[3] is emerging a provider of standards for interoperable clouds, a cloud management architecture, and a cloud service reference architecture.

3. OW2 Projects for Cloud Computing

This section presents the cloud computing functionalities of some existing OW2 projects. The presentation is structured around OSCi domains.

We retained the following list of the functions/qualities/features to perform the presentation.

– Elasticity: refers to the ability to quickly scale out and conversely release resources in order to meet user demand. The main resources that are concerned are CPU and storage.

– Reliability: refers to the ability to maintain the functioning of the cloud service under unexpected circumstances.

– QOS support and tools, metering, billing: refers to the ability to provide facilities that can enforce a predefined contract and level of agreement between the user and the platform provider. Refers also to the ability to measure the amount of cloud resources used with respect to the economical terms of use of the cloud platform.

3 http://www.dmtf.org/standards/cloud

– Agility: refers to the ability to rapidly change the cloud application to meet user requirements.

– Adaptability: refers to the capacity to add or remove functionalities.

– Security: refers to the level of security and confidentiality that can be guaranteed to end users.

– Tooling: refers to the API, programming model and development tools that are available for designing and implementing cloud applications and systems.

– Energy management: refers to the way power consumption can be measured and saved in a cloud infrastructure.

This document focuses explicitly on OW2 solutions. Yet many other non-OW2 solutions are available. Some of these products are complementary to OW2 solutions and could be integrated to provide a wider coverage of requirements for cloud solutions. The identified tentative list of non-OW2 solutions that fit in this category follows: OpenNebula, OpenStack, Eucalyptus, Nimbus, Apache (RedHat), Open cloud.org (IBM being the leader), RackSpace, Amazon EC2, Amazon S3, Google, Micorsoft Azure, Nirvanix, Linode, Euca2ools, S3CMD, DeltaCloud, NiftyName.

3.1) Domain 1: Self-sizing and green PaaS

Three OW2 projects are reviewed in this domain: JonAS, ProActive and Entropy. All three solutions are distributed under the terms of the LGPL license.

Other non-OW2 solutions that fit in the objectives of Domain 1 and that complement the previously reviewed approaches have been identified. This includes Xen, KVM, OpenStack, OpenNebula, Ubuntu.

JOnAS

JOnAS, JASMINe and CLIF are three independent OW2 projects that complement each others to form a consistent open source solution for cloud computing.

JOnAS is a leading edge Java EE 5 certified Open Source OSGi Enterprise Server. JASMINe aims to develop an administration tool dedicated to Java EE, MOM and SOA distributed applications. CLIF is a load injection framework.

- Elasticity
  Cluster Scaler for Java EE application elasticity (IaaS agnostic).

- Reliability
  Fail-over, session replication, error recovery.

- QoS support and tools, metering, billing
  Yes KPI, advanced monitoring for performance tracking and error detection, audit for metering application component and provides billing data.

- Agility
  Through its OSGi architecture, JonAS and JASMINe provides a highly dynamic

platform.

- Adaptability
  The OSGi modular architecture ensures an 'à la carte' deployment to fit to the application needs.

- Security
  Through Java EE standards.

- Tooling
  Advanced tooling are provided off-the-shelf; diagnostic tools (connection leak detector, classloaders explorer, bottleneck detector) , administration tools for production with JASMINe (automatic deployment, autonomic monitoring and error recovery).

- Energy management
  The dynamic platform and its on-demand service loading capabilities enables to optimize the resource consumption.

ProActive

ProActive is an innovative open source solution for parallel, distributed, multi-core computing.

- Elasticity
  On demand / On required resource (CPU) acquisition.

- Reliability
  Configurable fault tolerance and automatic restart the tasks and processes.

- QoS support and tools, metering, billing
  Management of priority using policies, built-in accounting and monitoring, metric available in a DB for integrating other BI and accounting tools.

- Agility
  Automatic acquisition of needed resources in function of any criteria (load, ...). Multiprotocol accessibility for distributed services with automatic selection of the best-fit resources. Private, Hybrid, Community and Public clouds. Transfer of data and applications between private and public instances.

- Adaptability
  Plugin based mechanism for adding new kind of resources and new scheduling policies. Support for Virtualization: VMware, Server/ESX, XenServer, VirtualBox, KVM/Qemu/Xen, Hyper-V.

- Security
  Configurable authentication mechanism (File based, LDAP...) and secured communications (SSL, SSH). Execution under User Id.

- Tooling
  Graphical, command line and web interfaces for managing and monitoring resources. Java, C++, WS, and REST API.

- Energy management
  Linux and Windows Agent for automatic resource acquisition, like idle desktops or servers, during off-peak hours (i.e. cycle scavenging or server harvesting).

Entropy

Entropy is a virtual machine manager for clusters which performs a globally optimized placement of virtual machines according to cluster resource usage and scheduler objectives.

- Elasticity
- Reliability
- QoS support and tools, metering, billing
- Agility
- Adaptability
- Security
- Tooling
- Energy management

OpenStack

OpenStack is a collection of open source technologies delivering a massively scalable cloud operating system. OpenStack is currently developing two interrelated projects: OpenStack Compute and OpenStack Object Storage.

- Elasticity
  Multi VM applications and nodes deployment

- Reliability
  “Rescue” mode allows an instance to mount affected disks and fix problem.

- QoS support and tools, metering, billing

- Agility

- Adaptability
  VM mobility will be supported in release Cactus in Apr 2011.

- Security

- Tooling
  Pause, suspend, lock, and password reset instances

- Energy management

OpenNebula

OpenNebula is an open-source (license Apache 2.0) project aimed at building the industry standard open source cloud computing tool to manage the complexity and heterogeneity of

distributed data center infrastructures.

- Elasticity
  Deployment of multi-tier services consisting of groups of inter-connected VMs; auto-configuration at boot time; and support for Microsoft Windows and Linux machine images

- Reliability
  Fault tolerance mechanisms for physical hosts and VMs, persistency of the OpenNebula daemon service. Few points of failure.

- QoS support and tools, metering, billing
  Support for accounting, billing can be build upon it. No QoS support for the moment.

- Agility
  Powerful and flexible requirement/rank matchmaker scheduler; and workload and resource-aware allocation policies such as packing, striping, load-aware, affinity-aware…

- Adaptability
  Enable the deployment of any cloud architecture: private, public, hybrid and federated; customizable plug-ins to access virtualization, storage, information, authentication/authorization and remote cloud services; new plug-ins can be easily written in any language; configuration and tuning parameters to adjust behavior of the cloud management instance to the requirements of the environment and use cases; and hook mechanism to trigger administration scripts upon VM state change

- Security
  Configurable authentication mechanism (File based, LDAP, SSH, easily extendible to others). Secured communications to worker nodes (SSH). Secure cloud interface through SSL or AWS auth mechanisms. Logging of any operation by user.

- Tooling
  Web GUI onemc. From 2.2, Sunstone, web interface for admins and users of the private cloud. XMLRPC API to manage resources lifecycle. Powerful CLI. OpenNebula Cloud API for third-party applications integration. Cloud interfaces (OCCI and EC2).

- Energy management
  Support of green scheduling policies through accounting and extensible policies in the scheduler.

3.2) Domain 2: Massively distributed services

Three OW2 projects are reviewed in this domain: Petals, ProActive and FraSCAti. All three solutions are distributed under the terms of the LGPL license.

Other non-OW2 solutions that fit in the objectives of Domain 2 and that complement the previously reviewed approaches have been identified. This includes OpenStack, OpenNebula.

Petals

Petals ESB is an open source ESB for large SOA architectures. Associated with Petals

Master and Geasy, they form a suite for cloud computing.

- Elasticity
  Cluster of Petals nodes (simple allocation mechanism).

- Reliability

- QoS support and tools, metering, billing
  SLA framework (WS-Agreement) provided by Petals Master + WSDM monitoring layer.
-
- Agility
  Web based BPMN 2.0 designer (GeasyBPMN) and BPEL conformance to generate JBI and BPEL artefacts.

- Adaptability
  Integration facilities provided by the bus allow to connect in-house information system (private cloud) and external services (public cloud).

- Security
  Confidentiality is provided by the messaging layer.

- Tooling
  Petals ESB: service container. Petals Master: Service governance platform. GeasyBPMN: BPM tooling.

- Energy management

ProActive

ProActive is an innovative open source solution for parallel, distributed, multi-core computing.

- Elasticity
  On demand / On required resource (CPU) acquisition.

- Reliability
  Configurable fault tolerance and automatic restart of the task/processes and application checkpoints.
- QoS support and tools, metering, billing
  Management of priority using policies, built-in accounting and monitoring, metric available in a DB for integration other BI and accounting tools.

- Agility
  Automatic acquisition of needed resources in function of any criteria (load, ...). Multiprotocol accessibility for distributed services with automatic selection of the best-fit resources. Private, Hybrid, Community, and public clouds. Transfer of data and applications between private and public instances.

- Adaptability
  Plugin based mechanism for adding new kind of resources , new scheduling policies

and new communication protocols. Support for Virtualization: VMware, Server/ESX, XenServer, VirtualBox, KVM/Qemu/Xen, Hyper-V.

- Security
  Configurable authentication mechanism (File based, LDAP...) and secured communications (SSL, SSH). Execution under User Id.

- Tooling
  Graphical command line and web interfaces for managing, monitoring resources and applications. Java, C++, WS, and REST API.

- Energy management
  Linux and Windows Agent for automatic resource acquisition, like idle desktops or servers, during off-peak hours (i.e. cycle scavenging or server harvesting).

FraSCAti

FraSCAti is an open source implementation of the SCA standard for large-scale service-oriented architectures. FraSCAti can be used both a standalone application server or as a service engine embedded in the Petals enterprise service bus.

- Elasticity
  Based on Petals nodes (Petals service engine).

- Reliability

- QoS support and tools, metering, billing

- Agility

- Adaptability
  Reconfiguration API for application-level services.

- Security
  JAAS enabled standard for services.

- Tooling
  Eclipse based development environment.

- Energy management
  Adaptation mechanism based on energy resource acquisition via Linux ACPI.

3.3) Domain 3: BI4Cloud

Three OW2 projects are reviewed in this domain: Talend, SpagoBI, BonitaSoft.

UshareSoft is a non-OW2 solution that fits in the objectives of Domain 3 and that complements the previously reviewed approaches.

3.4) Domain 4: Massively distributed cloud

TBC