mOSAIC
Facts, objectives
and current results

Dana Petcu

Institute e-Austria Timisoara
& West University of Timisoara
Romania
Overview

1. The problem: vendor-lock-in in Clouds
2. Target: portable applications
3. Objectives
4. Architecture
5. The role of open-source platforms
6. Further steps
Vendor lock-in, due to proprietary APIs

How to protect my application from lock-in?
Interoperability and portability: challenges

Scenarios using multiple Clouds

Federation of Clouds

Market

Multi-dimensional problem

POLICY:
Federate, communicate between providers

RUNTINE:
Migration support

DESIGN:
Abstract the programmatic differences
# Interoperability and portability: approaches

## Levels

<table>
<thead>
<tr>
<th>Network</th>
<th>Techs &amp; infrastr</th>
<th>Image &amp; data</th>
<th>Appl &amp; service</th>
<th>Management</th>
<th>Semantic</th>
<th>Business</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-deployment, work-loads</td>
<td>Protocols for requests/responses</td>
<td>Standards in deployment &amp; migration</td>
<td>Automation, configuration</td>
<td>Function calls and responses</td>
<td>Strategies, regulations, mode of use</td>
<td>E.g.</td>
</tr>
<tr>
<td>Allocation, admission</td>
<td>Protocols for requests/responses</td>
<td>Standards in deployment &amp; migration</td>
<td>Automation, configuration</td>
<td>Function calls and responses</td>
<td>Strategies, regulations, mode of use</td>
<td>E.g.</td>
</tr>
</tbody>
</table>

## Techs

- **Domain specific lang.**
  - Automated translation in code
- **Semantic repositories**
  - Mediators, frameworks (SLA@SOI)
- **Abstraction layers**
  - OVF/DMTF, CDMI/SNIA
- **Standards**
  - OCCI, Deltacloud
- **Protocols**
  - OpenStack, jClouds, libcloud, OpenStack
- **APIs**

E.g.**
PaaS – the worst case!

- each PaaS provider offers a special flavor in its design
- not all the features that are expected
  - rarely debug facilities,
  - no Security-as-a-Service
  - often not for private clouds
- portability is possible only between a small no. PaaSs
  - in case of open-source clones of the proprietary ones
- the problem escalates with the increase of no. PaaSs
  - increased no. in last two years
Opens-source & PaaS

Platform Service | Hosting | Integrated solution

- Well knows examples:
  - Google’s AppEngine, Microsoft’s Azure, RackSpace’s CloudSites, Amazon’s Beanstalk, Salesforce’s Force.com, Joyent’ SmartPlatform

- Appl support – different approaches:
  - Deploy code to specific VMs (Azure, Beanstalk)
  - Develop using rules, platform deal with deployment (AppEngine, Heroku)
  - Create metadata to be interpreted by PaaS at run-time (Force.com, OrangeScapes)

- Open-source only to develop apps
  - to allow customization

Platform Software | Software service | Deploy-based solution

- Deployment of middleware in data centers
- Easy way to deal with portability and interoperability (framework category)
- Open-source have the potential to impact the market as...
  - PVM/Parallel
  - Globus/Grid
mOSAIC: Open source API & Platform for multiple Clouds
marketing motto: “Flying through the Clouds”

1. a tool for developing portable Cloud-applications which can consume hardware and software resources offered by multiple Cloud providers;
2. an open-source PaaS that can be easily deployable by service providers and which can be customized and enhanced by service providers;
3. a brokerage system to support the decision of Cloud service provider selection at the deployment stage.
mOSAIC as R&D collaboration effort

Consortium:
1. Second University of Naples, Italy
2. Institute e-Austria Timisoara, Romania
3. European Space Agency, France
4. Terradue SRL, Italy
5. AITIA International Informatics, Hungary
6. Tecnalia, Spain
7. Xlab, Slovenia
8. University of Ljubljana, Slovenia
9. Brno University of Technology, Czech Republic

www.mosaic-cloud.eu

September 2011: 1st API implement. (Java)
September 2012: 1st stable PaaS, 2nd API impl. (Python)
March 2013: Full software package
Layered architecture

Open-source and deployable PaaS

Cloud adaptors
- Hosting services support
  - Amazon
  - Flexiscale
  - CloudBurst
  - GoGrid
  - VMware
  - Rackspace
  - CloudSigma
  - NIIIFII
  - Arctur
  - Hostko

Deployable services support
- Eucalyptus
- OpenNebula
- DeltaCloud
- OpenStack

Other Cloud hosting, deployable services

mOSAIC PaaS and IaaS

API implementations
- Java & Python cloudlets
- Java connectors (KV, MQ, DFS, HTTPgw)
- Python, Erlang, Node.js connectors (KV, MQ)
- Demo applications

Application support

Application tools
- Eclipse plug-ins
- Frontends (cmdl, web)
- Backends (WS)
- Configuration tools
- PortableTestbedCluster

Semantic engine
- Semantic query builder
- Pattern builder
- Reasoner
- Maintainer
- Search engine
- Ontologies

Software platform support

Platform’s core components
- Register & Discover
- Packager & Deployer
- Provisioner & Monitor
- Operate & Maintain
- Scheduler & Scaler
- Interoperability support
- mOS

Application service components
- SLA components
- Benchmark component

Application support components
- Deployable COTS: RabbitMQ, CouchDB, MySQL, HTTg
- Drivers: AMQP, Riak, HDFS, S3

Infrastructure support

Cloud Agency
- Mess. Transfer Protocol
- Mediator
- Meter
- Archiver
- Tier agents

Agents for Cloud Agency
- Broker
- Vendor agents

Cloud-enabled applications

mOSAIC’s proof-of-the-concept applications

- Earth Observation applications
- Intelligent maintenance system
- Information extraction
- Model exploration service
- Analysis of structures
- User community developed applications

OS repository:
https://bitbucket.org/mosaic
Usage scenario:

- **Write component-based application**
  - Languages: Java, Python, NodeJS, Erlang
  - Communications through message passing
  - Respect the event-driven style of programming
  - Find the proper functionalities with the Semantic Engine

- **Debug your application on the desktop or on-premise server(s)**
  - Within Eclipse
  - Use Personal Testbed Cluster using VirtualBox for the VMs

- **Deploy your application in a Cloud**
  - Assisted by Cloud Agency and Broker (with SLAs)
  - Provisioner of the platform if you have already credentials

- **Monitor & modify the applications**
  - Control the life-cycle of the components (start/stop/replace)
## Open-source Platform Software

<table>
<thead>
<tr>
<th>Product</th>
<th>AppScale</th>
<th>Cloud Foundry</th>
<th>ConPaaS</th>
<th>mOSAIC</th>
<th>OpenShift</th>
<th>TyphoonAE</th>
<th>WaveMaker</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Owner</strong></td>
<td>Univ. California</td>
<td>VMWare</td>
<td>Contrail Consortia</td>
<td>mOSAIC Consortia</td>
<td>RedHat</td>
<td>Tobias Rodäbel</td>
<td>VMWare</td>
</tr>
<tr>
<td><strong>Languages</strong></td>
<td>Python, Java, Go</td>
<td>Java, Ruby, Node.js, Groovy</td>
<td>PHP</td>
<td>Java, Python, Node.js, Erlang</td>
<td>Java, Python, Perl, PHP, Ruby</td>
<td>Python</td>
<td>Java</td>
</tr>
<tr>
<td><strong>Data Support</strong></td>
<td>HBase, Redis, Hypertable, MySQL Cluster, Cassandra, Voldemort, MongoDB, Memcached-DB</td>
<td>MongoDB, SQLFire, Postgresql, Redis</td>
<td>Scalaris, MySQL, XtreamFS</td>
<td>Riak, CouchDB, HDFS, MemcacheDB, Redis, MySQL</td>
<td>MySQL, MongoDB, Amazon RDS</td>
<td>MongoDB, MySQL, Berkeley DB JE</td>
<td>Amazon S3, Rackspace</td>
</tr>
<tr>
<td><strong>OS</strong></td>
<td>Ubuntu, CentOS on Xen, KVM</td>
<td>VMWare image</td>
<td>XtreemOS image</td>
<td>Linux, deploy mOS on top</td>
<td>Red Hat Virtualization</td>
<td>Debian, Ubuntu</td>
<td>VMWare image</td>
</tr>
<tr>
<td><strong>Messaging</strong></td>
<td>Channel</td>
<td>RabbitMQ</td>
<td>Own design</td>
<td>RabbitMQ</td>
<td>Own design</td>
<td>RabbitMQ, ejabberd, Channel</td>
<td>Own design</td>
</tr>
<tr>
<td><strong>Clouds tested</strong></td>
<td>Amazon EC2, Eucalyptus</td>
<td>VMWare</td>
<td>Own testbed</td>
<td>Amazon EC2, Eucalyptus, OpenNebula, Flexiscale ...</td>
<td>RightScale Rackspace, Smart-Cloud, Amazon</td>
<td>Google</td>
<td>EC2, Rackspace, OpSource, Eucalyptus</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>CLI, Web</td>
<td>CLI</td>
<td>Web</td>
<td>CLI, Web, REST</td>
<td>CLI, REST</td>
<td>CLI</td>
<td>Studio</td>
</tr>
</tbody>
</table>
# Open-source Platform Software

<table>
<thead>
<tr>
<th>Product</th>
<th>CloudFoundry</th>
<th>mOSAIC</th>
<th>OpenShift</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Development support</strong></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Dedicated to web aps or general</td>
<td>Web apps</td>
<td>General</td>
<td>Web apps</td>
</tr>
<tr>
<td>Desktop Cloud Simulator</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>API access</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Thread access</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Allows to choose stack components</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Debugging mode</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Deployment support</strong></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Lock-in when building own Cloud</td>
<td>Yes (VMWare)</td>
<td>No</td>
<td>Yes (RHE)</td>
</tr>
<tr>
<td>Web server (e.g. Tomcat)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Build-in-balancer</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Performance analytics</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Support multiple Cloud providers</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Agreements SLA</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Deploy with a special tool</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Support Private Cloud</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Allows to add third party components</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Execution support</strong></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Command line (CLI)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Web console</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Access to logs via web</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Web based monitoring</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Multitenant</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
What’s next?

- Full integration & testing & benchmarking & promotion of mOSAIC PaaS - deadline Spring 2013
- Develop further the open-source and deployable PaaS: Improve platform services (e.g. new on-going projects)
- Use the platform and programming style for real applications