Cloud powered services composition using Public Cloud PaaS platform

Student: Andrey Afanasyev
Supervisor: Yuri Demchenko

Graduate School of Informatics
University of Amsterdam

3 February 2015
What is Cloud Computing

Definition according to NIST SP 800-145 The NIST Definition of Cloud Computing:

“Cloud computing is a model for enabling ubiquitous, 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. This cloud model is composed of five essential characteristics, three service models, and four deployment models.”

Cloud Computing Definition: Components

1. Five basic Cloud characteristics
   1. On-demand self-service
   2. Broad network access
   3. Resource pooling
   4. Rapid elasticity
   5. Measured Service

2. Three basic service models
   1. Software as a Service (SaaS)
   2. Platform as a Service (PaaS)
   3. Infrastructure as a Service (IaaS)

3. Deployment models
   1. Private clouds
   2. Public clouds
   3. Hybrid clouds
   4. Community clouds
Relation between IaaS, PaaS, SaaS

Packaged Application (Hosted)

Cloud IaaS

Cloud PaaS

Cloud SaaS

Data

Application
Runtime
Middleware
OS
Virtualisation
Servers
Storage
Networking

Data

Application
Runtime
OS
IaaS Mgmt Platf
IaaS Platform
Virtualisation
Servers
Storage
Networking

Data

Application
Apps Dev Platf
PasS Platform
Runtime
OS
Virtualisation
Servers
Storage
Networking

Data

Application
SaaS Platform
Runtime
OS
Virtualisation
Servers
Storage
Networking

Source: IEEE Cloud Course, Tutorial 1
Research questions

**Primary**
How cloud can improve design-deployment-test-improvement business application development cycle?

**Secondary**
What are criteria and key performance indicators (KPI)?
Approach

1. Installation and configuration of the chosen software on a private and enterprise-oriented cloud
2. Stress testing and analysing results of the both environments
3. Interpreting and providing conclusions of acquired results
Private Cloud (Old Style | Hosted)

1. Hardware based on london.studlab.os3.nl server (Dell PowerEdge R210):

<table>
<thead>
<tr>
<th>ITEM</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel XeonCPU L3426 @ 1.87GHz</td>
</tr>
<tr>
<td>VGA</td>
<td>MGA G200eW WPCM450 (226MB)</td>
</tr>
<tr>
<td>RAM</td>
<td>DDR3 1066MHz ECC 8192 MB (7966 available)</td>
</tr>
<tr>
<td>HDD</td>
<td>WDC WD5002ABYS-1 500GB</td>
</tr>
</tbody>
</table>

1. Software:
   1. Server OS: Ubuntu Server 14.04.1 LTS
   2. Virtualisation software: KVM/QEMU 2.0.0
   3. WebPanel: Nginx, WebVirtMgr
   4. OpenERP (ODOO 8) as a SaaS
Why OpenERP (ODOO 8)?

- Multitenancy
  - An important feature of cloud computing
- Three-tier architecture
- Developed in Python

Source: http://en.wikipedia.org/wiki/Multitier_architecture#Three-tier_architecture
Why OpenERP (ODOO 8)?

Source: https://doc.odoo.com/6.0/developer/1_2_module_development/1_server_module/
Measurement Results (Jmeter)

150 users (login)

All request are processed.
# Measurement Results (Jmeter)

## 200 users(login)

<table>
<thead>
<tr>
<th>Label</th>
<th># Samples</th>
<th>Average</th>
<th>Median</th>
<th>90% Line</th>
<th>Min</th>
<th>Max</th>
<th>Error %</th>
<th>Throughput</th>
<th>KB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOAP/XML-RP...</td>
<td>200</td>
<td>40591</td>
<td>42518</td>
<td>59165</td>
<td>9827</td>
<td>61725</td>
<td>2.00%</td>
<td>3.2/sec</td>
<td>2.2</td>
</tr>
<tr>
<td>TOTAL</td>
<td>200</td>
<td>40591</td>
<td>42518</td>
<td>59165</td>
<td>9827</td>
<td>61725</td>
<td>2.00%</td>
<td>3.2/sec</td>
<td>2.2</td>
</tr>
</tbody>
</table>

- **Increased number of dropped request**

---

**Aggregate Graph**

- The graph shows increased time metrics for the 200 users login test.
Microsoft Azure

• Why Microsoft Azure?
  • Virtual Hardware (350 cores per subscription)
  • Platform Development possibilities (.NET, Python, JavaScript etc...)
  • Amount of services and Applications
  • Ongoing SURFmarkt trial project with Microsoft Azure
    • UvA is provided with free account and virtually "unlimited" resources

• Microsoft Azure Cloud services model:
  • IaaS
  • PaaS
Microsoft Azure PaaS

WEBSITES + WEBHOSTING

Deploy and scale modern websites and web apps:
- .NET, Java, PHP, Node.js, Python
- Built-in AutoScale and Load Balancing
- High Availability with Auto-patching

SQL DATABASE

A relational database-as-a-service that allows to access a tier-1 capabilities:
- SQL Databases
- MySQL
- MongoDB

Service Bus
- SOA based applications integration
- Messaging service
- Queue
- Namespace and Access Control
Microsoft Azure Platform Layers

3 functional layers

- Microsoft Azure core services that include Compute, Storage, and Connect network connectivity service
- SQL Azure that include Database service and additionally Reporting and Data Synchronisation
- AppFabric applications fabric that provides a number of services for service integration such as Service Bus, Access Control and Caching.
Microsoft Azure Architecture (details)

Windows Azure Service Architecture

THE INTERNET VIA TCP OR HTTP

Web Role
IIS as Host

Load Balancer (LB)

Worker Role
Managed Interface Call

Storage

Tables
CACHE
Blobs

Service Bus Relay, Queue, Broker

Windows Azure Data Center
Scalable Application Archetype ("pay-as-you-go")

Intelligent Network Load Balancer

Network Activation

Stateless Web and/or Application Servers

Stateless ‘Worker’ Machines

Async Activation

State Tier | Queues | Key/Value Datastores | Partitioned RDBMS | Shared Filesystem
Service Bus Queue

Service Bus queues are a general-purpose technology that can be used for a wide variety of scenarios:

Communication between
- Web and worker roles in a multi-tier Microsoft Azure application.
- On-premises apps and Microsoft Azure hosted apps in a hybrid solution.
- Components of a distributed application running on-premises in different organizations or departments of an organization

Improves isolation between application tiers and layers

Allows fine granular control over the messages properties (message filtering)
Problem

• OpenERP (ODOO 8) is not a part of Microsoft Azure PaaS

Solution

• Install a website with a SQL server (Virto Commerce)
• Configure Scaling and related instances (Website Dashboard)
Virto Commerce

• Part of Websites Services
• ASP.NET (C#)
• Azure SQL
Scale

• Configuration
  • Standard Plan
  • Large Instance
  • Instance count 10
Monitoring

CPU Time: 1170.50 s
Data In: 63.38 MB
Data Out: 364.33 MB
Http Server Errors: 0
Requests: 19786

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOURCE</th>
<th>MIN</th>
<th>MAX</th>
<th>AVG</th>
<th>TOTAL</th>
<th>ALERT RULES</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Time</td>
<td>andrey1virto</td>
<td>0 ms</td>
<td>584.67 s</td>
<td>21.68 s</td>
<td>1170.50 s</td>
<td>Not Configured</td>
</tr>
<tr>
<td>Data In</td>
<td>andrey1virto</td>
<td>0 B</td>
<td>18.53 MB</td>
<td>1.17 MB</td>
<td>63.38 MB</td>
<td>Not Configured</td>
</tr>
<tr>
<td>Data Out</td>
<td>andrey1virto</td>
<td>0 B</td>
<td>83.38 MB</td>
<td>6.75 MB</td>
<td>364.33 MB</td>
<td>Not Configured</td>
</tr>
<tr>
<td>Http Server Errors</td>
<td>andrey1virto</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Not Configured</td>
</tr>
<tr>
<td>Requests</td>
<td>andrey1virto</td>
<td>0</td>
<td>4646</td>
<td>366.41</td>
<td>19786</td>
<td>Not Configured</td>
</tr>
</tbody>
</table>
Measurement Results (Jmeter)

<table>
<thead>
<tr>
<th>Label</th>
<th># Samples</th>
<th>Average</th>
<th>Median</th>
<th>90% Line</th>
<th>Min</th>
<th>Max</th>
<th>Error %</th>
<th>Throughput</th>
<th>KB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Page</td>
<td>2000</td>
<td>42929</td>
<td>42005</td>
<td>50020</td>
<td>34534</td>
<td>349663</td>
<td>0.05%</td>
<td>5.4/sec</td>
<td>192.6</td>
</tr>
<tr>
<td>TOTAL</td>
<td>2000</td>
<td>42929</td>
<td>42005</td>
<td>50020</td>
<td>34534</td>
<td>349663</td>
<td>0.05%</td>
<td>5.4/sec</td>
<td>192.6</td>
</tr>
</tbody>
</table>

2000 samples is stable limit to not get a Out of Memory Error by Jmeter on the client machine

An order increase in # processed request
Lessons Learned: Development Environment and Tools

• Tools
  • WebMatrix3
  • Visual Studio 2013 Community (PHP, Python, C# etc...)
  • Windows Azure SDK

• Deployment Slot

• Publishing Profiles

• Source Control (DropBox, GitHub, VisualStudio, CodePlex etc.)
Conclusion

<table>
<thead>
<tr>
<th>ADVANTAGES OF CLOUD COMPUTING</th>
<th>DISADVANTAGES OF CLOUD COMPUTING</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Cost Efficiency</td>
<td>1. Control and Reliability</td>
</tr>
<tr>
<td>2. Unlimited* Compute Power and Storage</td>
<td>2. Security, Privacy and Compliance</td>
</tr>
<tr>
<td>3. Backup and Recovery</td>
<td>3. Interoperability, Compatibility and vendor lock</td>
</tr>
<tr>
<td>4. Easy Access</td>
<td></td>
</tr>
<tr>
<td>5. Global Availability and geo localisation</td>
<td></td>
</tr>
<tr>
<td>6. Quick Deployment</td>
<td></td>
</tr>
<tr>
<td>7. Availability of VM/Apps Marketplace (or VM Depot @ Azure)</td>
<td></td>
</tr>
</tbody>
</table>
Questions?