Mixing Public and private clouds

a Practical Perspective

Maarten Koopmans
Nordunet Conference 2009
Who

• ING Group - 2002
• SURFnet 2002-6
• ICTU (govt) 2006-8
• vrijheid.net 2008-
• qtask.com
• ibeamsystems.com
What

- What is a cloud?
- Cost model
- Mixing private and public clouds
- Cloud and Grid
- Time left? Details, details...
Confusion
"Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS), so we use that term. The datacenter hardware and software is what we will call a Cloud.

When a Cloud is made available in a pay-as-you-go manner to the public, we call it a Public Cloud; the service being sold is Utility Computing. Current examples of public Utility Computing include Amazon Web Services, Google AppEngine, and Microsoft Azure. We use the term Private Cloud to refer to internal datacenters of a business or other organization that are not made available to the public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not normally include Private Clouds."

— Above the clouds paper, Berkeley
So...

- Pay as you go
- Public
- Utility computing + SaaS
Cost model

- No up front cost

- Pay per resource - GB storage, transfer, computing hour

- Discount when reserving e.g. virtual machines; drives total cost down for a small upfront investment
Cost model

It’s OK to read sheets up front. Just remember this one!

Given a parallelizable computing job of $N$ hours the cost of $N$ computers running 1 hour equals the cost of 1 computer running $N$ hours.
Cost Example

🌟 2 servers, 2 months, 50GB test data during development

🌟 8 servers, 2 months, 1TB data running time

<table>
<thead>
<tr>
<th></th>
<th>Monthly cost</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>$162</td>
<td>$324</td>
</tr>
<tr>
<td>Running time</td>
<td>$786</td>
<td>$1572</td>
</tr>
<tr>
<td>Total cost</td>
<td></td>
<td>$1896</td>
</tr>
</tbody>
</table>
Opportunities – part I

(Managers might want to make a note)

★ Use public cloud model to model private clouds

★ Cross-NREN clouds, scale up, cost goes down

★ Public clouds, NREN service (like e.g. certificate service)
What have we done?

Amazon Web Services

Eucalyptus

Google App Engine
Google App Engine, Eucalyptus

Two students: π, MPI
Lessons learnt
(time to make a note again)

• Google App Engine is really well suited for certain web applications. *Web applications.*

• Eucalyptus works, sort of. Promising, but the amount of time to get an infrastructure working on par with AWS is too much. *Expect this to change within a year.*
Your data, backup and restore

Public

AWS, Eucalyptus

Metadata, queues

Data storage

Private

Eucalyptus

Your data, backup and restore
AWS, Eucalyptus

Public

Metadata, queues

Private

Data storage, image processing

Process images in parallel
Lessons learnt

• It’s all about resource sharing!!!
• Credential management is key
• Queues are everywhere, the rest may differ
Opportunities II

- Resource sharing!
- Credential management across clouds
- Scaling out (private -> public)
- Billing and metering in cross-NREN scaling out (this is a BIG one)
Cloud+Grid

Or
Cloud+Grid

★ Storage: as back end, public data sets or directly

★ Buffer computing capacity – remember the computing hour equivalence rule!

★ “a simpler solution than GRID”
Questions?

Now......

or on to the details

(or later......
maarten@vrijheid.net)
Cloud components

Everything is there........

...for a reason!
Cloud components

- Computing on demand (virtual machines), IP provisioning, block device service (persistent hard disks), load balancing and service monitoring

- Storage Service

- Simple large tables

- Queuing system

- Map-reduce
Simple Example within Amazon

EC2
Virtual machine(s)

Image booting

EBS
"Virtual Disk"

Image bundling

S3
Simple Storage Service
Example data upload
Example: data restore

SimpleDB: filename, size, date, s3-us, s3-eu, state(hot, transit, cold), lasttouched

EC2 instance

Download data fill

File download queue

Customer file target

S3 file download worker

S3 US
Example: data processing

- SimpleDB:
  - filename, size, date, s3-us, s3-eu, state(hot, transit, cold), lasttouched

- EC2 instance (dispatch)
  - Download data location, add job

- Job processing
  - queue

- EC2 instance (worker)
  - Worker, fetch data from S3
  - Put result in S3

- S3 US
Even more questions?

Now

or......

maarten@vrijheid.net