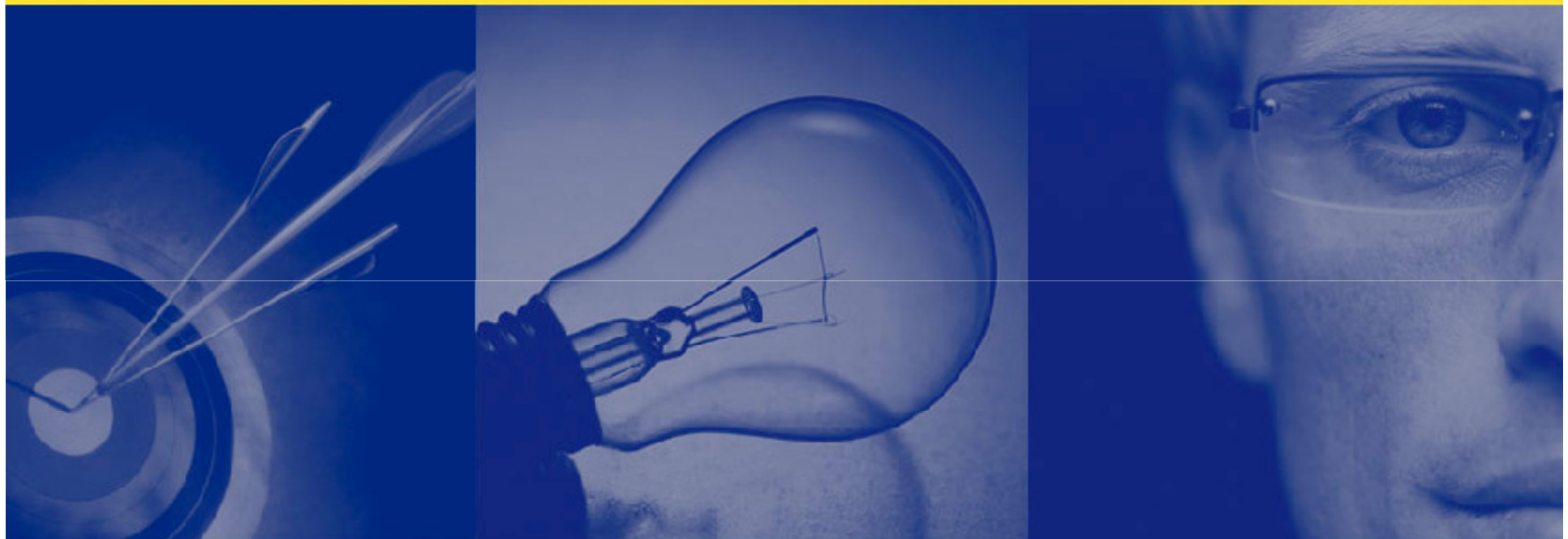


Using Cloud Services for Test Environments

A case study of the use of Amazon EC2



Lee Hawkins (Quality Architect)
Quest Software, Melbourne



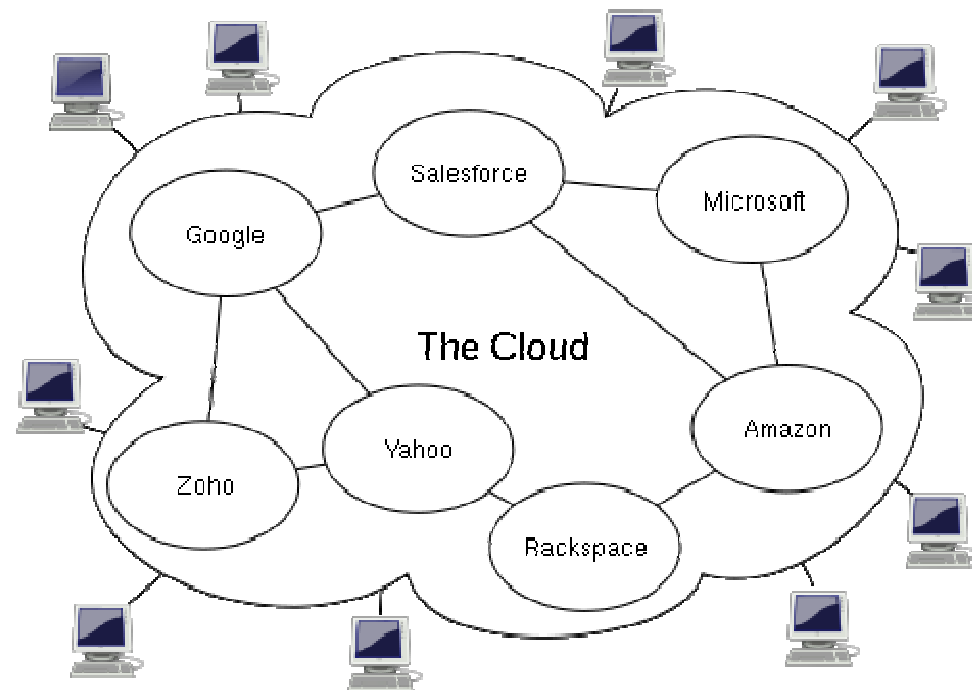
We are gathered here today...

... to talk about:

- Some cloud computing basics, terminology, major players, NoSQL databases, etc.
- The reasoning behind our use of Amazon EC2 as a testing infrastructure
- The good
- The bad

Cloud computing basics (part one) – what's the cloud?

- Shared resources, software, and information are provided on demand over the Internet.
- In a public cloud, these resources are dynamically provisioned on a self-service basis over the Internet, via web applications/web services, from an off-site third-party provider who bills on a fine-grained utility computing basis.





Cloud computing basics (part two) - SaaS

- In **Software as a service**, the idea is to deliver software products as a service over the Internet:
 - Network-based access to, and management of, commercially available (i.e. not custom) software - commonly CRM and accounting packages
 - Removing the need to install & run the application on your own hardware
 - Simplifying maintenance & support
 - Centralized updates, so no more downloading patches and performing upgrades
- A major cloud service provider in this space is **Salesforce** (CRM).



Cloud computing basics (part three) - PaaS

- In **Platform as a service**, the idea is to provide both the computing platform and solution stack as a service over the Internet:
 - Instead of just offering applications in the form of SaaS, vendors in this space offer an entire platform because they provide the foundation upon which to build highly scalable and robust Web-based custom applications.
- The major players in this space include **Amazon** (Web Services), **Google** (App Engine) and **Microsoft** (Windows Azure).



Cloud computing basics (part four) - IaaS

- In **Infrastructure as a service**, the idea is to deliver computing infrastructure (typically a virtualization environment) as a service:
 - Instead of purchasing servers, data centre space or network equipment, you instead buy those resources as a fully outsourced service. Capital expenditure is converted to operational expenditure.
 - Typically billed on a utility computing basis.
 - Easy deployment and low cost of entry.
 - Highly scalable and highly available.
- The major players in this space are **Amazon** (**Elastic Compute Cloud, EC2**) and **Rackspace** (**Cloud Servers**).



Cloud computing basics (part five) – NoSQL databases

- Process huge volumes of data in the Cloud. The volume of data produced by a web site like Twitter is too enormous to be processed by a conventional RDBMS. The current daily message volume of Twitter is around 7 TB.
- Also commonly referred to as “No SQL” databases because their interface is not native to SQL. To query, access and modify the data requires writing code (to access their APIs) that can be tedious, expensive and time consuming even for software developers to write. This lack of familiar SQL access is a barrier to use for less technical consumers of the data.



Cloud computing basics (part six) – NoSQL databases

- Some examples of significant users of the more common No SQL databases:
 - (Apache) Cassandra – Twitter, Facebook, Digg
 - (Amazon) SimpleDB – Amazon, Netflix
 - (Open) MongoDB – Foursquare , bit.ly, sourceforge
 - (Apache) HBase – StumbleUpon, Mahalo
 - (Microsoft) Azure – VeriSign, bing



Our testing challenge (part one) - new products

We were tasked with rapidly testing a couple of new Quest tools coming into the cloud database market:

Toad for Cloud Databases - tool for putting “No SQL” databases (such as Azure Table Services, Cassandra, SimpleDB and HBase) within the reach of RDBMS developers and DBAs, by enabling them to use SQL to:

- Work with data in a Cloud database
- Join a Cloud database to a relational database
- Copy tables from one Cloud database to another or between a Cloud database and a relational database

OraOop - a connector between Oracle and Apache Hadoop for fast and scalable data transfer between Oracle databases and Hadoop.



Our testing challenge (part two) – test environment

The biggest challenge in testing these tools was the creation of a decent test environment.

What kind of requirements did we have of this environment?

- Availability of large number of Windows and Linux servers.
- Ability to tear down and spin up with minimal effort.
- Freedom to use potentially huge amounts of disk space.
- Limited timeframe for setup.

Our options:

- Acquire and configure physical hardware,
- Leverage in-house ESX virtualized environment, or
- Leverage an IaaS offering

Our testing challenge (part three) – we go with EC2

Use in-house physical test environment



- Too slow to acquire hardware and configure
- Too costly for the number of machines we needed
- Too inflexible, we needed to ramp up/down the number of machines in the environment at the push of a button

Use in-house ESX virtualized environment



- Limited scalability and concerns about resource allocation
- Potential to impact performance of other critical users on the ESX cluster during “heavier” testing

Use IaaS offering



- We chose Amazon EC2...



The good news (part one)

- Very easy to spin up new Windows and Linux machines of varying specifications.
- Very easy to ditch the environment and set up again – only paying for the time it was up.
- We configured a high-end Oracle database machine with TBs of storage for performance testing in a very short period of time. When we needed more storage, it was easy to add on.
- Good I/O bandwidth between Amazon instances.



The good news (part two)

- The Hadoop distribution we were testing against (provided by Cloudera) included scripts to deploy Hadoop on a cluster of EC2 instances out of the box - this saved massive amounts of environment setup time (perhaps person weeks).
- Availability of huge range of Amazon Machine Images (AMIs), templates covering many common OS's, databases, applications, etc. - including (importantly for us) those from Cloudera.
- High availability – we never experienced any downtime during our testing with EC2.



The bad news (part one)

- Limited tooling - the only viable management tools are:
 - ElasticFox (Firefox plug-in),and
 - Toad for Cloud Databases EC2 Console
- Hard/impossible to know exactly what hardware you're getting:
 - EC2 instance types allocate CPU resources based on “virtual cores” and EC2 “compute units”. How are these defined?
 - Each of the EC2 instance types has an I/O performance indicator (*low*, *moderate*, or *high*). How many I/Os can we expect at each level?
 - While this fulfils the purpose of making it easy to compare CPU and I/O capacity between different instance types within Amazon EC2, the lack of precision makes it hard to know what you're buying and also difficult to compare against other IaaS providers.



The bad news (part two)

- Cost blowout is a risk
 - Amazon bill monthly “post paid”, so there is potential for “bill shock”
 - Limited tooling to keep track of accumulated costs
 - “Low” per hour rates accumulate rapidly when deploying large environments and leaving them running
- Limited set of Amazon Machine Images in the APAC region.
- There may be significant hurdles in using such test environments if regulation, compliance (e.g. SOX), data sovereignty, and security are issues for your organization and/or its (test) data.
- Accessibility of your test environment is totally dependent upon both your internet connection and the availability of the IaaS provider.



Summary

- The use of Amazon EC2 suited our test environment requirements very well and saved a great deal of setup and maintenance time.
- We found the EC2 environment to be reliable.
- It is unlikely that we would have been able to achieve the same level of test coverage by using physical hardware or ESX virtual hardware given the tight timeframes.
- Our test data was of no intrinsic value, but there are clearly compliance and related issues that might be barriers to adoption for other products/organizations with sensitive data.

How to find out more...



Any questions?

lee.hawkins@quest.com

- Amazon EC2 – <http://aws.amazon.com/ec2/>
- NoSQL databases - <http://nosqlpedia.com/>
- Windows Azure case studies - <http://www.microsoft.com/windowsazure/evidence/>
- Cloudera – <http://www.cloudera.com>
- Quest Software – <http://www.quest.com>
- Quest’s “Toad for Cloud Databases” (free beta) & “OraOop” (freeware) - <http://toadforcloud.com/index.jspa>

Nobody said it was going to be easy, and nobody was right. **George Bush**
Any fool can criticize, condemn, and complain - and most fools do. **Dale Carnegie**