lucid

Cloud Computing: A Brief Summary

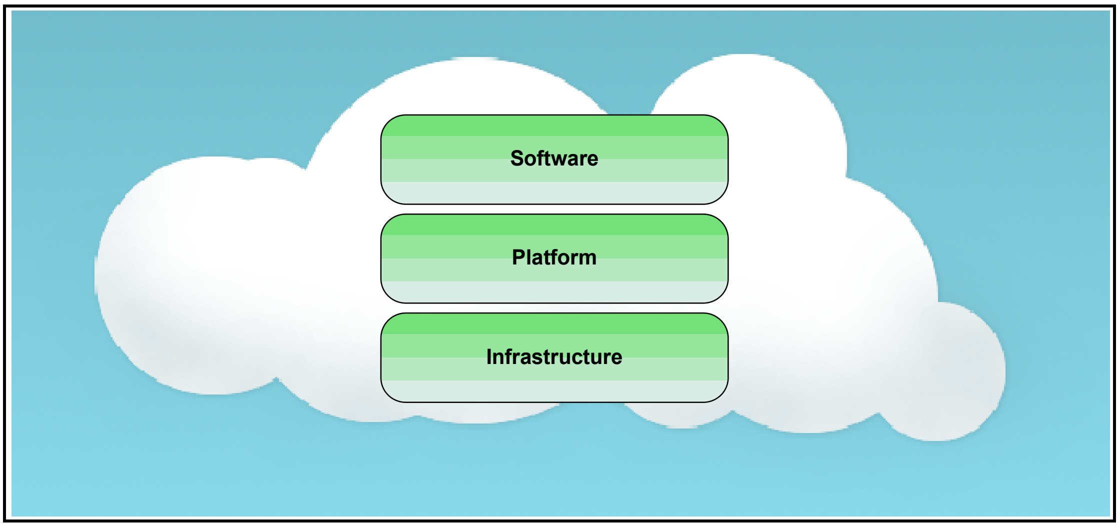
Lucid Communications Limited
Prepared by Neil Turner
September 2009

I. Contents

1. Introduction

Cloud Computing is a broad and ill-defined term, but in essence it amounts to virtualised third-party hosting. That is, rather than renting part or all of an actual physical server from a hosting company, you rent a certain amount of server *resources*. Your server runs inside a virtual container which can be moved from one physical server to another without interruption of service. Such a container is also capable of spanning multiple physical machines, giving it potentially limitless resources.

A web server typically has three tiers to it: The physical infrastructure, the operating system platform, and the web application software being run. A cloud container may contain one, two or all of these layers:



Choosing an appropriate cloud provider is often a matter of selecting which layers you wish to control yourself, and which should be the responsibility of the hosting provider.

In brief, the options are these:

- If you want a fully operational server with total control over the operating system and configuration, you want **Infrastructure as a Service**.
- If you don’t want any configuration hassles at all, if you just want to upload your files and leave your website to run, you want **Software as a Service**.
- For something in between, with a pre-configured operating system but a reasonable amount of control over the rest of the configuration, **Platform as a Service** is for you.

It is important to understand that the requirement that drove the development of Cloud Computing was scalability. With dedicated or shared servers, an increase in traffic can leave the site badly underpowered. This is a major obstacle to the development of new mass-market web applications, since upgrading to better hardware is a difficult and time-consuming process, but start-ups generally can’t afford powerful server farms from the start.

For companies looking to host internal or specialist web applications, scalability is not as much of a priority; cost is a much bigger factor, and one that is not necessarily emphasised by cloud providers. Nevertheless, cloud options do exist that have very low baseline costs, which could mean savings for low-volume sites. In addition, because cloud systems are virtualised there are a number of natural advantages: backups are much easier to create and the risk associated with hardware failure is minimised.

2. Software as a Service (SaaS)

"Software as a Service" has of late become something of a marketing buzzword, and is inevitably misused, but there seem to be two types of offering that fall into this category.

The first is virtualisation of internal systems – a large company's CRM or ERP is moved from intranet servers into the cloud and hosted on shared servers. This applies both to custom-written systems and commercial applications like Microsoft SharePoint. There are a number of companies that specialise in single-application hosting of this type, and a useful directory to these is available here.

The second type is the simplest form of cloud computing: effectively virtualised shared-server space into which you put anything you want. This *may* be an internal business application, but it could equally be a public-facing website.

This type, works almost exactly the same way as regular shared server hosting. The host gives you access to a virtual directory to which you can upload a single website's files, but any changes to the platform or environment have to be handled by the hosting company. For multiple web applications, you need multiple SaaS virtual sites, which are charged separately but can all be managed though a single web console.

So how is SaaS better than an ordinary shared server? With SaaS, the website exists in a virtual container, which can be moved from physical server to physical server without interruption of service. This makes it much easier to transfer the site to more powerful hardware as the need arises, but all of the other limitations of shared servers still apply: very limited configurability, no third-party software or background services, and so on.

2.1. Disadvantages

Most of the disadvantages of SaaS are the same as those that apply to ordinary shared hosting; the relative severity of the disadvantages will of course depend on your specific circumstances.

Much of the security burden falls on the hosting company. You would have to trust them to properly secure the server(s) on which the virtualised application runs, as well as the firewall controlling network security.

You would have no control over what other applications are sharing the same server; each other application provides a potential point of entry for hackers, and poorly written software might have adverse effects on the stability of the operating system. This is slightly mitigated by virtualisation; in the event of server instability, the hosting company could easily transfer the site to a different server. You would however be reliant upon them to monitor their systems closely and expedite the move.

Audit trails and service monitoring are more difficult to achieve in a virtualised environment, since you would have no direct access to the system logs. If you need log access, you would have to configure your application to handle it internally.

2.2. Advantages

The biggest advantage of SaaS, as with all cloud computing, is its scalability. Virtualised applications can be moved onto different hardware extremely quickly in response to increased demand. SaaS also shares the other standard advantages: easy backups, flexible pricing, portability, and so on.

Compared to other forms of cloud computing, SaaS is extremely user-friendly, with the hosting company taking most of the maintenance burden.

SaaS solutions also have the potential to be extremely cheap if you have simple requirements and don't expect a huge amount of traffic.

2.3. Example: Rackspace Mosso

Rackspace are one of the really big names in traditional hosting, so it makes sense to consider their new cloud-based offering: Mosso. Their basic package (priced in US dollars) starts at $1200 per year. For this price, you get:

- 500GB of bandwidth per month
- 50GB of hard disk space
- 10 000 compute cycles[1] per month.

It's a little peculiar that this is their basic offering; a typical site would need tens of thousands of monthly visitors to get anywhere near the 500GB limit, so one might have thought there would be a more limited (and therefore cheaper) option available.

All the same, $100 per month does look like good value if you are expecting high traffic volumes for your site, and the incremental cost of going beyond the standard package (50c per GB of storage, 25c per GB of bandwidth, 1c per compute cycle) is attractively low. You can also get a 100MB SQL Server database for $5 per month extra.

For basic starter sites, however, there are better options out there.

2.4. Example: WebFusion

If you're looking to host a small site, WebFusion seems a decent option. Their basic package is priced at just £48 per year, for which you get 1.5GB of disk space and 20GB of traffic per month. However, at the moment this offering only supports ASP.NET 1.1.

For sites that require .NET 2.0/3.5 and a SQL server database, the cost would be a still-reasonable £180 per year. This includes 10GB of disk space and 150GB of traffic.

[1] This is roughly equivalent to a 2.8GHz processor running 24 hours a day for a month.
Rackspace claim that this is enough to serve up 11 million page views of their own website.

3. Infrastructure as a Service (IaaS)

For all the convenience of hosted web applications, there is a lot to be said for the control that comes with having access to the entire server, from the operating system on up. In cloud computing, this is possible through Virtual Private Servers.

With a Virtual Private Server, all that is abstracted away is the hardware. Logging in to a VPS is practically indistinguishable from logging in to a remote server via Terminal Services. If you have a dedicated server in a hosting centre somewhere, moving to VPS should leave you with an almost identical experience.

Virtual Server technology is not new. Developers have been using it for years to set up test platforms for new applications, or to subdivide physical machines into multiple logical servers. Using a virtual machine to host a web application was uncommon, however, because the virtual server would by definition have to be less powerful than the physical server on which it was running. What makes IaaS interesting is that this limit no longer applies: a VPS can now be shared across multiple physical servers, limited only by network latency.

VPS is available through a number of companies, including WebFusion, CloudNine (Hosting.com), Joyent, Nirvanix and ParaScale. The biggest player in the market, though, is Amazon.

3.1. Disadvantages

One potential problem is that Virtual Private Servers are typically incapable of passing a PCI Security Audit, making them unsuitable for high-security functions like handling credit card information. One could not, therefore, use a VPS to host an e-commerce site.

In addition, there is the issue of loss of control. Providers like Amazon reserve the right to shut off the server without prior notice if it is behaving in a way that leads them to believe it has been compromised by hackers, or if they think we are using it for unethical activities like spamming. This means that if you were to end up on a blacklist by mistake, the consequences would be worse than with a non-cloud server.

3.2. Advantages

VPS machine images can be saved as a file which can be deployed as a virtual server later. This means that one effectively has a “clean install” of the required environment available at all times as a rollback position in the even of a catastrophic failure.

Moreover, the state of the running virtual server can be “saved” at any time, providing a very convenient backup procedure, albeit one that would be difficult to test without incurring charges from the provider (the server image would be in a proprietary format, so the only way to test would be to pay them to launch an instance of it).

The basic cost of a VPS server appears to be slightly lower than that of an equivalent dedicated platform. As with all cloud solutions, cost scales according to use, so costs come down if traffic levels fall, which is very useful for anyone whose revenue stream depends on traffic.

3.3. Example: Amazon EC2

With Amazon Elastic Compute Cloud (EC2), you need to set up an Amazon Machine Image – a virtual machine running Windows (or Linux) onto which you install all the applications, libraries, and data that you need.

You then create a running instance of the Image, which acts like an ordinary server. You can log in via Terminal Services and administer the server in the same way as for a remote physical server.

The server’s capabilities will scale automatically according to conditions you define. This means that you can effectively place an upper boundary on costs.

Amazon throw in a monitoring system, CloudWatch, which lets you keep track of the server’s resource usage (and therefore your running costs).

The EC2 pricing model is incredibly complicated, but its basic package – one processor, 1.7GB of RAM, 160GB of storage – is available for as little as £700 per year, and a larger instance – four processors, 7.5 GB of RAM, 850GB storage – works out to £2800 per year. Unfortunately, these options do not include a SQL Server licence.

The pricing of EC2 servers jumps significantly when SQL server is added, nearly doubling. This makes EC2 platforms potentially very expensive if you need the full version of SQL server. SQL Server Express Edition, which is free but excludes many features, may be an option in some cases.

4. Platform as a Service (PaaS)

Platform as a Service is a compromise between SaaS and IaaS. The real difference is that instead of having space on a Windows-based server, you use a specifically-written cloud platform, which you access through a web interface, and you build your site using tools specific to that platform. This platform is considerably more configurable than an SaaS site, but at the same time doesn’t require all of the maintenance and administration of a full IaaS VPS.

The major PaaS platforms also compare favourably to SaaS in terms of the extra services they offer – a variety of vendor-supplied tools and SDKs to assist in development and maintenance.

PaaS is the newest category of cloud hosting options, and is likely to be the most fiercely competitive market, with Microsoft and Google going head-to-head.

4.1. Disadvantages

Vendor lock-in is always a concern when it comes to PaaS. One would have to write applications to be tailored to the chosen platform, and migrating an application out of that platform onto a standard dedicated server would be a problem.

As with IaaS, full compliance with security standards like PCI would be a problem.

All PaaS platforms share the disadvantage that there is a limit to the options available in terms of third-party applications. This is mainly caused by the relative novelty of the platforms, which means very few developers have released compatible versions of their software at this stage. Both Microsoft Azure and the Google Application Engine require developers to port their software to the new platform. It will take some time before the full range of software currently available on a dedicated Windows Server becomes available on PaaS.

4.2. Advantages

Any PaaS solution comes with the advantage of minimising the developer's maintenance time while still providing a considerable amount of customisation and configuration.

There is also an argument to be made that since the two biggest players in the industry – Microsoft and Google – are investing so heavily in PaaS cloud computing, there is a certain inevitability to their emergence as a standard. This means that the odds of useful third party tools being developed for PaaS systems are very high.

4.3. Example: Google App Engine

Google App Engine is ideal for relatively simple public-facing websites. Google go to great lengths to make their systems easy to use, and their pricing is extremely cheap – in fact, it's free for the first gigabyte of storage and 1.3 million HTTP requests.

To build a Google App Engine website, you set up an account and get access to the App Engine Software Development Kit (SDK), which allows you to write App Engine-compatible software. Unfortunately (and unsurprisingly since this is Google), the platform does not support Microsoft Technologies; Python or Java must be used.

GAE is unfortunately very clearly intended for quite small applications and simple websites. The platform enforces hard limits on the amount of processing your application can do – any request that takes longer than 30 seconds to complete or that sends more than 10MB of data is immediately terminated.

If you like to use Microsoft technologies, look elsewhere, but if you're technology-agnostic and have a relatively simple website, the GAE may be for you.

4.4. Example: Microsoft Windows Azure

Azure is Microsoft's equivalent of the GAE, and as you'd expect it is slightly more focused on the use of .NET components (it natively supports ASP.NET). It is also set up to support SQL Server databases (via the Azure Data Services system), which would significantly reduce the difficulty of migration for people used to Windows-based servers.

The Azure platform appears to be a modified form of Windows Server 2008; you get access to many of the usual technologies, including Internet Information Services (IIS), although this access would be via a web interface rather than Remote Desktop. Azure also supports non-Microsoft development languages including PHP and Python, which increases your options somewhat.

Azure uses an entirely pay-as-you-go pricing system, charging 12c per hour plus 15c per gigabyte of data stored (per month) plus 15c per gigabyte transferred. That leads to an estimated annual cost (for the equivalent of a mid-range server) of $1120, or around £700 (fluctuating with the exchange rate). This pricing does not include the separate licensing for SQL server, but unlike other suppliers

Microsoft are able to offer this at very little extra cost. A gigabyte of database space costs $120 per year, though Microsoft do not make it clear whether this is for a logical database *server*, or just for a single database.

Microsoft intends in the near future to offer SharePoint via Azure, stretching it into the SaaS market. In the long run, they also intend to create a full Virtual Private Server (IaaS) version of Azure.

Using Azure would simplify the deployment process for ASP.NET developers, since Microsoft have apparently devoted a lot of time to ensuring that Azure deployment is possible “at the click of a button” directly from Visual Studio.