Cloud Computing Models

Eugene Gorelik

Working Paper CISL# 2013-01

January 2013

Composite Information Systems Laboratory (CISL)
Sloan School of Management, Room E62-422
Massachusetts Institute of Technology
Cambridge, MA 02142

Cloud Computing Models

by

Eugene Gorelik

Submitted to the MIT Sloan School of Management and the MIT Engineering Systems Division
in Partial Fulfillment of the Requirements for the Degrees

of

Master of Management and

Master of Engineering

in conjunction with the System Design and Management Program

at the

Massachusetts Institute of Technology

January 2013

Signature of Author ....

MIT Sloan School of Management

MIT Engineering Systems Division

Certified by ....

Stuart Madnick

John Norris Maguire Professor of Information Technologies &
Professor of Engineering Systems, Thesis Supervisor

Accepted by ....

Patrick Hale

Director, System Design and Management Program

Cloud Computing Models

Comparison of Cloud Computing Service and Deployment Models

by

Eugene Gorelik

Submitted to the MIT Sloan School of Management and the MIT Engineering Systems Division in Partial Fulfillment of the Requirements for the Master Degree in Engineering and Management

ABSTRACT

Information Technology has always been considered a major pain point of enterprise organizations, from the perspectives of both cost and management. However, the information technology industry has experienced a dramatic shift in the past decade – factors such as hardware commoditization, open-source software, virtualization, workforce globalization, and agile IT processes have supported the development of new technology and business models.

Cloud computing now offers organizations more choices regarding how to run infrastructures, save costs, and delegate liabilities to third-party providers. It has become an integral part of technology and business models, and has forced businesses to adapt to new technology strategies.

Accordingly, the demand for cloud computing has forced the development of new market offerings, representing various cloud service and delivery models. These models significantly expand the range of available options, and task organizations with dilemmas over which cloud computing model to employ.

This thesis poses analysis of available cloud computing models and potential future cloud computing trends.

Comparative analysis includes cloud services delivery (SaaS, PaaS, IaaS) and deployment models (private, public, and hybrid). Cloud computing paradigms are discussed in the context of

technical, business, and human factors, analyzing how business and technology strategy could be impacted by the following aspects of cloud computing:

- Architecture
- Security
- Costs
- Hardware/software trends (commodity vs. brands, open vs. closed-source)
- Organizational/human Factors

To provide a systematic approach to the research presented in this paper, cloud taxonomy is introduced to classify and compare the available cloud service offerings.

In particular, this thesis focuses on the services of a few major cloud providers. Amazon Web Services (AWS) will be used as a base in many examples because this cloud provider represents approximately 70% of the current public cloud services market. Amazon's AWS has become a cloud services trendsetter, and a reference point for other cloud service providers.

The analysis of cloud computing models has shown that public cloud deployment model is likely to stay dominant and keep expanding further. Private and Hybrid deployment models are going to stay for years ahead but their market share is going to continuously drop. In the long-term private and Hybrid cloud models most probably will be used only for specific business cases. IaaS service delivery model is likely to keep losing market share to PaaS and SaaS models because companies realize more value and resource-savings from software and platform services rather than infrastructure. In the near future we can expect significant number of market consolidations with few large players retaining market control at the end.

Thesis Supervisor: Stuart Madnick

Title: John Norris Maguire Professor of Information Technologies & Professor of Engineering Systems

TABLE OF CONTENTS

Contents

1 INTRODUCTION

1.1 Historical background

The idea of providing a centralized computing service dates back to the 1960s, when computing services were provided over a network using mainframe time-sharing technology. In 1966, Canadian engineer Douglass Parkhill published his book *The Challenge of the Computer Utility* [1], in which he describes the idea of computing as a public utility with a centralized computing facility to which many remote users connect over networks.

In the 1960s, the mainframe time-sharing mechanism effectively utilized computing resources, and provided acceptable performance to users; however, mainframes were difficult to scale and provision up-front because of increasingly high hardware costs. Accordingly, users didn't have full control over the performance of mainframe applications because it depended on how many users utilized the mainframe at a given moment. As such, with the introduction of personal computers users loved the idea of having full control of their computing resources, even though these resources are not as effectively utilized.

With the change in the semiconductor industry, personal computers became affordable, and business abandoned mainframes. A new challenge was then introduced: how to share the data.

Client-server systems were supposed to address this data-sharing challenge by providing centralized data management and processing servers. As business computing needs grew and the Internet became widely adopted, the initially simple client-server architecture transformed into more complex two-tier, three-tier, and four-tier architectures. As a result, the complexity and management costs of IT infrastructure have skyrocketed – even the costs of actual software development in large organizations are typically lower than costs of software and infrastructure maintenance.

For many enterprises, the long-standing dream has been to background information technology issues and concentrate on core business instead. Although the effect of the cloud computing adoption is yet to be seen, many companies believe that cloud computing may offer feasible alternative model that may reduce costs and complexity while increasing operational efficiency.

1.2 Definition

There are countless definitions and interpretations of cloud computing to be found from multiple sources. The term "cloud computing" itself likely comes from network diagrams in which cloud shape are used to describe certain types of networks, either the Internet or internal networks. Some sources refer to cloud computing as a set of applications delivered as services combined with the datacenter hardware and software that enables the applications. Others say that cloud computing is a business model rather than a specific technology or service.

In our opinion, cloud computing consists of both technological and business components. Certain cloud-enabling technologies significantly helped to form the cloud, and it is unlikely that cloud computing could have existed without them. We discuss these more closely in the next chapter, but it is worth mentioning that cloud-enablers such as open-source software, virtualization, distributed storage, distributed databases, and monitoring systems are the cornerstones of cloud infrastructure.

Cloud computing assumes that every software application or system component becomes a service or part of a service. Therefore, the architecture of new or existing systems might have to be changed to become cloud compatible. As such, in order to realize the value of the cloud and enable it for an organization, businesses must typically make major structural adjustments to internal IT organizations and evangelize cloud philosophy to employees. Depending on the type of cloud used by an organization, this may also create competition within the company. It is typical that people resist change, so cloud evangelists often face resistance within their organizations.

1.3 Why Cloud Computing?

Let's consider a few of the most important factors that provide key incentives for organizations to use cloud computing.

1.3.1 Elasticity

The ability to scale computing capacity up or down on-demand is very important. For example, imagine a company that provides software-as-a-service (SaaS) online tax-filling services. Obviously with such a business model, this organization's computing resource demand will peak during tax season – only two to three months each year. Financially, it doesn't make sense to invest up-front knowing that computing infrastructure will remain only partially utilized nine or ten months per year.

1.3.2 Pay-As-You-Grow

Public cloud providers like Amazon allow companies to avoid large up-front infrastructure investment and purchase new computing resources dynamically as needed – companies needn't plan ahead and commit financial resources up-front. This model is particularly feasible for smaller companies and start-ups, which often cannot afford to spend large sums of money at the beginning of their business journey.

1.3.3 In-House Infrastructure Liability and Costs

Running information technology inside the company incurs substantial liability and costs. While some would argue that running infrastructure inside the organization is safer and cheaper, that's not necessarily the case. Depending on a company's IT budget, employee skills, and some other factors, it could worth running infrastructure from a public cloud. Public cloud providers could offer reasonable service-level agreements (SLA) and take care of the liability headaches that company CIOs may face.

1.4 Why Now?

Why is cloud computing happening only now, instead of many years ago?

1.4.1 Economies of Scale

The enormous growth of e-commerce, social media, and various Web 2.0 services has tremendously increased the demand for computational resources. Companies like Google, Amazon, and Microsoft quickly realized that financially it is more feasible to build very large data centers for their needs than many small ones – it is much more cost-efficient to buy resources like electricity, bandwidth, and storage in large volumes (see Table 1). In larger datacenters, it becomes easier to maximize the amount of work per dollar spent: you can share components in a more efficient way, improve physical and virtual server density, reduce idle server times, and cut administrator/server ratio.

	Medium-sized DC	Very Large DC	Ratio (Large-to-Small DC)
Network	**$95 per Mbit/sec/month**	**$13 per Mbit/sec/month**	**7.1**
Storage	**$2.20 per GByte/month**	**$0.40 per GByte/month**	**5.7**
Administration	**~ 140 Servers/ Admin**	**>1000 Servers/ Admin**	**7.1**

Table 1: Economies of scale in 2006 for a medium-sized datacenter (1000 servers) vs. a very large datacenter (50,000 servers). [2][3]

As shown in Table 1, the cost of network bandwidth and system administration is 7.1 times cheaper, and the cost of storage 5.7 times cheaper, in 50,000-server datacenters compared to datacenters with only 1000 servers.

1.4.2 Expertise

It takes lots of investment and technical know-how to build a datacenter. Some companies developed substantial expertise in that area. Once these companies built datacenters for their internal clouds, they realized that they could leverage existing expertise and technology to build public cloud datacenters and offer computing services to other companies. As a result, companies like Amazon and Google became public cloud providers. (See [27] How and why did Amazon get into Cloud Computing business by Verner Vogels, CTO, Amazon)

1.4.3 Commodity Hardware

Drops in the costs of computer chip production, architecture standardization around the x86 platform, and the increasing mechanical compatibility of internal PC components led to a significant decrease in computer hardware costs over the past decade. Hardware affordability has contributed to its commoditization, and accordingly reduced computational costs.

1.4.4 Virtualization

Hardware virtualization (see “2. Cloud Computing’ – a Definition”) has allowed increasing hardware utilization density, and ensures that hardware resources are utilized more efficiently. This is one of the technologies that enables elasticity, and so has provided increased flexibility in terms of speed of deployment, dynamic auto-provisioning, and cloud management.

1.4.5 Open-Source Software

Open-source software and commodity hardware are major cloud computing enablers. The Linux operating system in particular has become a major building block at the heart of largest cloud environments. Similarly, virtualization software Xen is used by Amazon to host the largest set of virtual machines in the world (approximately half-a-million as of now [26]), and Hadoop distributes a computing platform that helps thousands of companies to run massive parallel computations in the cloud (Amazon Elastic MapReduce service). The ability to avoid expensive software license costs is one of the factors that enables companies to provide affordable cloud services.

1.5 Cloud vs. Grid

Many experts would argue that cloud computing comes from grid computing. However, although there are many similarities between cloud and grid computing, these methodologies are not the same. The main difference is that grids were not originally created as a public on-demand utility computing service, and are typically used within the same organization to run heavy computational tasks. Cloud computing is instead normally associated with a specific service, and that service is used as an access point providing results to the service consumer – who might be a user or another application (a B2B application, for example).

Computational grids are historically used for large computational jobs and built with many servers up-front, while the advantage of the cloud is that it can be scaled on-demand. The cloud offers more elasticity, such that an environment can start from only a few servers, grow quickly to hundreds of servers, and then scale back down to the initial size if required.

2 “CLOUD COMPUTING” – A DEFINITION

This chapter discusses cloud computing technology and cloud models. As an example of a public cloud we consider Amazon Web Services (AWS), and for a private cloud VMWare cloud technology. These providers hold most of the market share in their specific niches, and are worth reviewing.

2.1 An Introduction to Cloud Architecture

As the introduction notes, the idea of providing centralized computing services over a network is not new – mainframe timesharing technology was popular as far back as the 1960s, but was replaced by personal computers and client-server architecture. Until around 10 years ago, typical enterprise computing infrastructure consisted of powerful and very expensive servers. Infrastructure architecture was monolithic, and each of these powerful machines could easily host 20-30 enterprise applications. This market was dominated by only a few hardware vendors, such as IBM, Sun, HP, and Dec, whose servers were expensive to purchase and maintain, took considerable time to install and upgrade, and in some cases were vulnerable to server outages that could last several hours until a vendor representative delivered proprietary replacement parts.

The operating system was installed directly to hardware, and most of the servers hosted multiple applications within the same operating system without providing physical or virtual isolation (see Figure 1). Because it was difficult to quickly move and rebalance applications across servers, server resources were not utilized most effectively.

Distributed applications, which were installed over multiple servers, communicated with each other using CORBA or DCOM communication protocols over RPC. However, it was a major problem with such protocols that they were vendor-dependent, and so the implementation of one vendor might not be compatible with that of others. This was solved at the beginning of the 2000s by the introduction of web services, which use open specifications that are language, platform, and vendor agnostic.

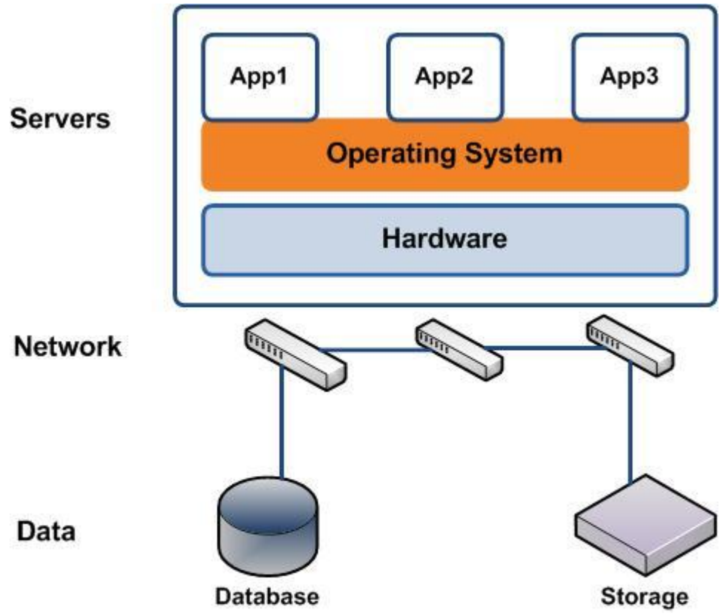


Figure 1: Servers without virtualization

With the introduction of virtualization, things have changed tremendously. Virtualization improves resource utilization and energy efficiency – helping to substantially reduce server maintenance overhead and providing fast disaster recovery and high availability. Virtualization has been very important for cloud computing, because it isolates software from hardware and so provides a mechanism to quickly reallocate applications across servers based on computational demands (see Figure 2).

Virtualization was a major step towards cloud infrastructure; however, the service component was still missing. Virtualized environments managed by internal system administrators and by default virtualization platforms do not provide the abstraction layer that enables cloud services. To cloud-enable an environment, a layer of abstraction and on-demand provisioning must be provided on top (see Figure 3). This service layer is an important attribute of any cloud environment – it hides the complexity of the infrastructure, and provides a cloud-management interface to users. Depending on the interface implementation, a cloud-management interface can be accessed through a management dashboard, REST or SOAP web services, programming APIs, or other services. For example, Amazon Web Services provides access through a management dashboard or REST/SOAP web services.

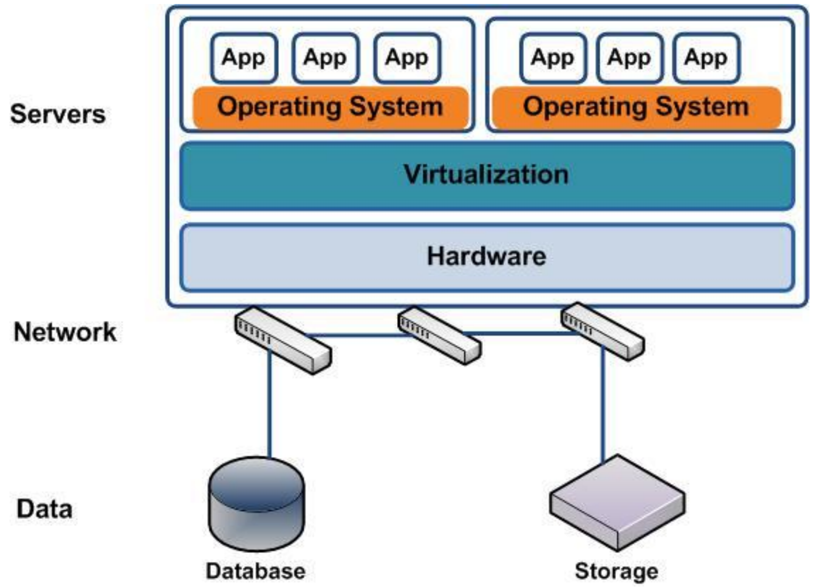


Figure 2: Virtualized servers

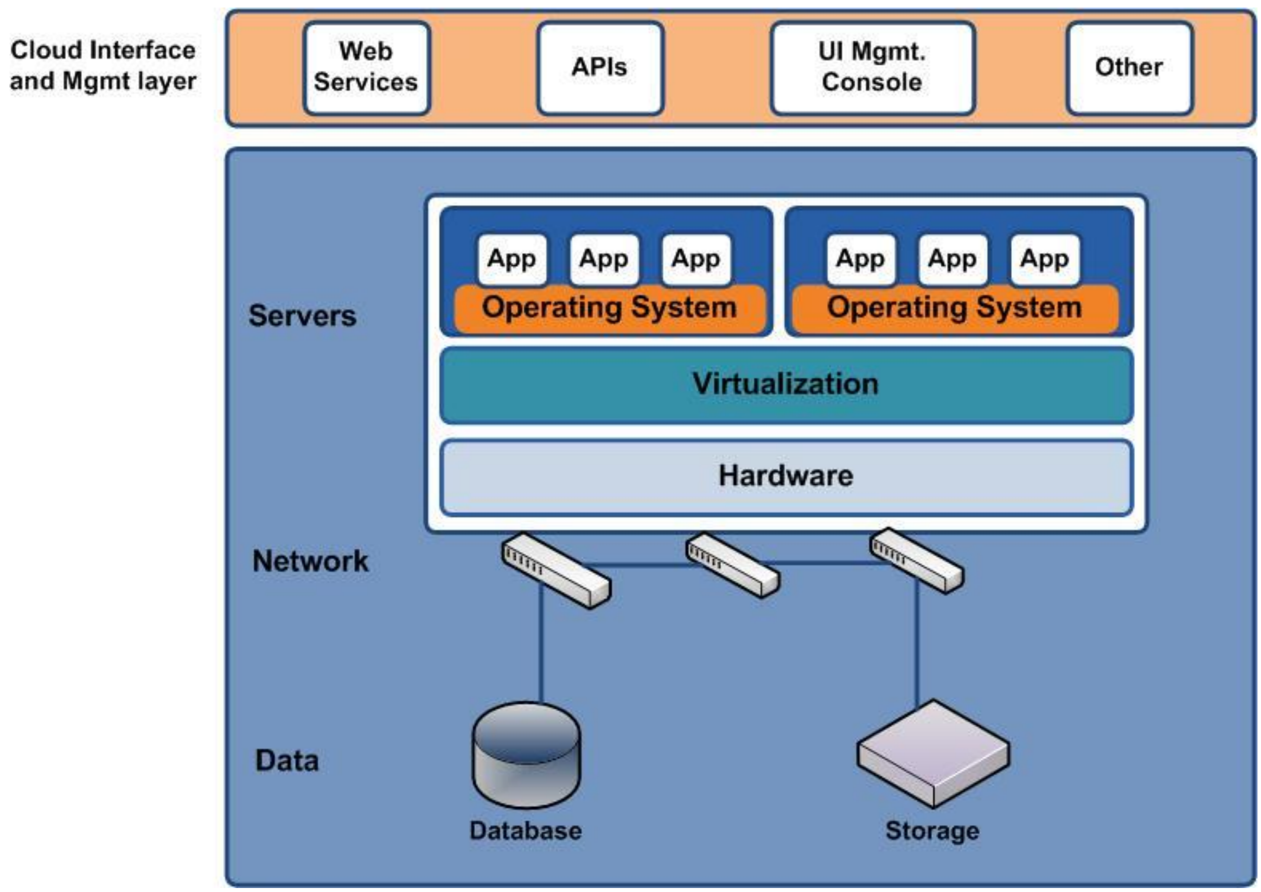


Figure 3: Simplified cloud infrastructure

Cloud management interfaces (for example, the Amazon admin console) provide functions allowing users to manage a cloud lifecycle. For instance, users can add new components to the cloud such as servers, storage, databases, caches, and so on. Users can use the same interface to monitor the health of the cloud and perform many other operations.

2.2 Cloud Services

The cloud can provide exactly the same technologies as "traditional" IT infrastructure – the main difference, as mentioned previously, is that each of these technologies is provided as a service. This service can be accessible over a cloud management interface layer, which provides access over REST/SOAP API or a management console website.

As an example, let's consider Amazon Web Services (AWS). AWS provides multiple cloud infrastructure services (see Figure 4) [4]:

Amazon Elastic Compute Cloud (EC2) is a key web service that provides a facility to create and manage virtual machine instances with operating systems running inside them. There are three ways to pay for EC2 virtual machine instances, and businesses may choose the one that best fits their requirements. An **on-demand** instance provides a virtual machine (VM) whenever you need it, and terminates it when you do not. A **reserved** instance allows the user to purchase a VM and prepay for a certain period of time. A **spot** instance can be purchased through bidding, and can be used only as long as the bidding price is higher than others. Another convenient feature of Amazon's cloud is that it allows for hosting services across multiple geographical locations, helping to reduce network latency for a geographically-distributed customer base.

Amazon Relational Database Service (RDS) provides MySQL and Oracle database services in the cloud.

Amazon S3 is a redundant and fast cloud storage service that provides public access to files over http.

Amazon SimpleDB is very fast, unstructured NoSQL database.

Amazon Simple Queuing Service (SQS) provides a reliable queuing mechanism with which application developers can queue different tasks for background processing.

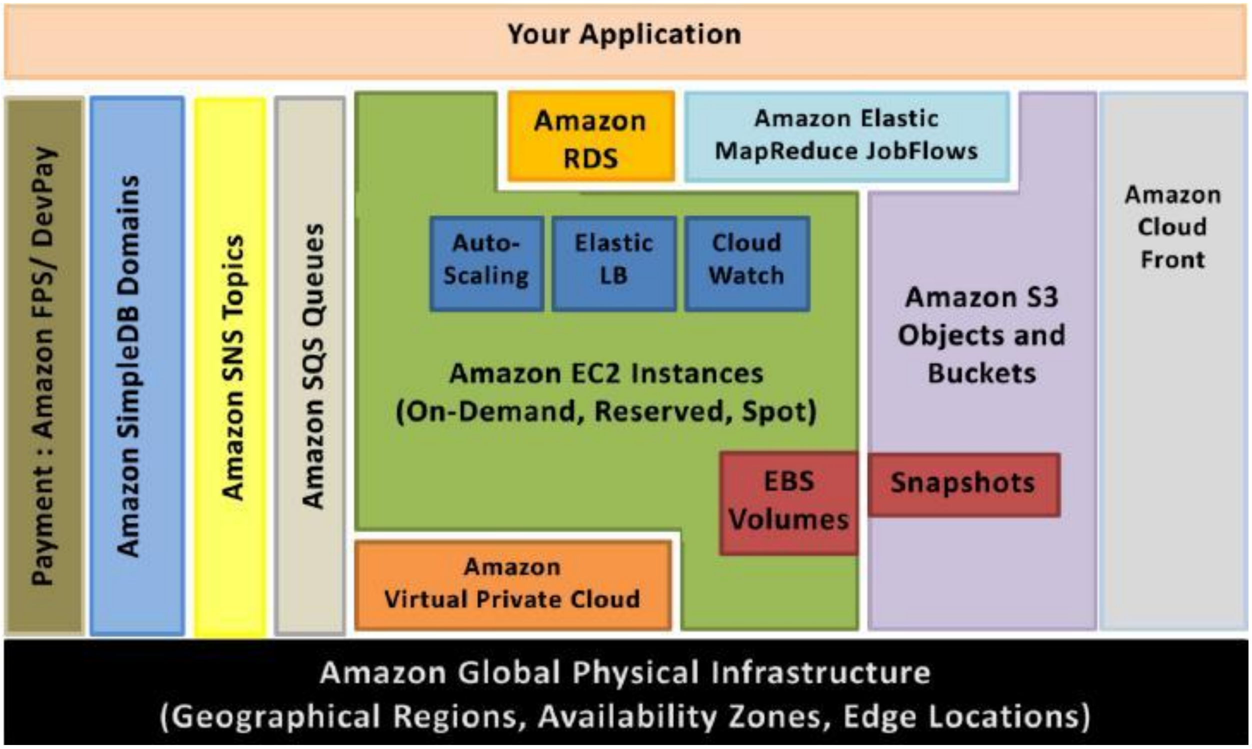


Figure 4: Amazon Web Services cloud [4]

Here we do not describe every single Amazon service, but you can see how massive and powerful Amazon's cloud presence is. We review some of these services in more detail later in this paper.

2.3 Building Scalable Architecture

One of the most important factors in infrastructure architecture is the ability to scale. In the "traditional" non-cloud infrastructure, systems are typically architected to sustain potential future growth and resource demand. Organizations must invest considerable financial resources up-front to provision for future growth. Because non-cloud infrastructures do not provide elasticity, system resources cannot quickly upscale and downscale; this leads to constant resource overprovisioning, and therefore systems are inefficiently underutilized most of the time. Conversely, cloud infrastructure is multi-tenant, and so computing resources are shared across multiple applications. This shared multi-tenant environment is based on the assumption that all hosted applications cannot normally be busy at the same time – when one application is idle, another is busy. This way, cloud providers can allocate resources on-demand and improve the efficiency of computing resources utilization. The cloud infrastructure consists of set of shared hardware resources – that is, servers, storage, and networks. Sophisticated cloud management software automatically monitors resource utilization and allocates these resources as needed. The cloud provider must ensure the availability of computing resources to serve all customers, even at peak times.

2.3.1 Horizontal Scaling vs. Vertical Scaling vs. Automated Elasticity

Vertical scaling (scale-up) assumes that organizations make substantial up-front investment and do not worry about available computing resources until the point when demand is approaching capacity limit (see “scale-up approach,” the blue dotted line in Figure 5). When this capacity is reached, company must invest substantially in expanding (see “Huge capital expenditure” in Figure 5). However, if company responds late and demand outgrows existing capacity, this may affect the business (see “You just lost your customers” in Figure 5). This vertical scaling model is normally seen in old-style non-cloud environments.

Horizontal scaling (scale-out) enables organizations to expand their environment in small chunks on-demand (see the maroon scale-out approach line in Figure 5). Horizontal scaling is a popular methodology because it provides a way to quickly scale-out resources without major capital expenditures; however, it still requires up-front capacity and demand planning.

Automated elasticity means that a cloud provider continuously monitors a customer’s infrastructure and scales it on-demand.

If you look at Figure 5, the black dashed line with an arrow at the end shows predicted demand. The further a company is from this demand, the more money they are wasting. The green line shows automated elasticity, which closely follows the “actual demand” red line. Automated elasticity provides the minimal possible deviation from predicted demand, and also minimizes customer-loss risk

Example: Company XYZ plans to purchase servers to host their web applications. They currently host serve 5000 users. XYZ predicts user growth of approximately 1000 users per month. These are their three choices:

1. **Vertical scaling**: purchase two powerful servers, $50,000/each. This should provide the capacity to host up to 50,000 users. When these servers reach capacity, XYZ will buy another two servers.
2. **Horizontal scaling**: purchase two servers, $3,000/ each, with the capacity to serve 10,000 users. When XYZ needs to serve more users, they will buy additional servers.
3. **Automated Scaling:** purchase the required computing resources from a cloud provider to serve 5,000 users, and let the cloud provider auto-scale the capacity.

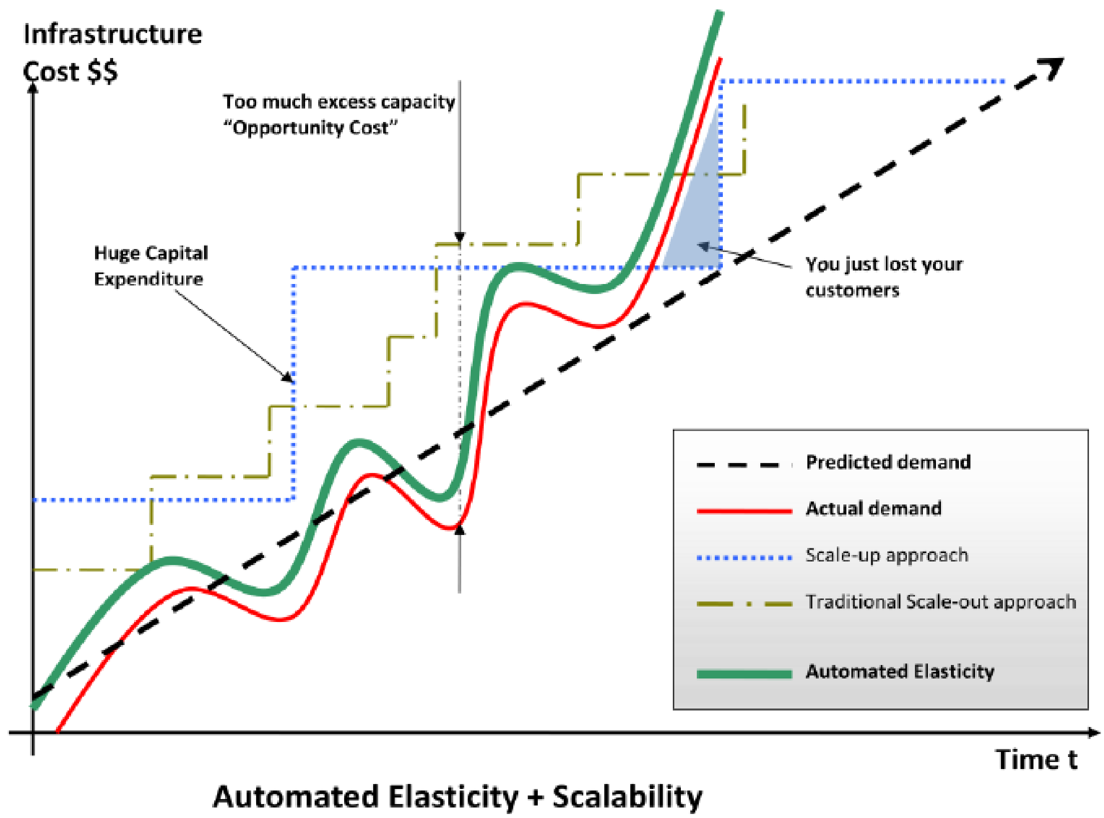


Figure 5: Automated elasticity and scalability [4]

2.3.2 Elasticity

Elasticity is a key benefit of the cloud. It allows dynamically-scaling computing resources based on actual demand. In the "traditional" infrastructure, organizations would have to predict computing resource demand up-front, while conversely the primary goal of cloud computing is to adjust computing resources automatically in response to demand (see Figure 5). The closer your resource utilization is to actual demand, the higher the cost efficiency of your cloud infrastructure.

Infrastructure elasticity potential strongly depends on how hosted software applications are modeled, and how infrastructure components are built. For example, it is difficult and expensive to scale-out complex monolithic applications that cannot be broken into smaller modules. To add more computing power for such non-modular applications, organizations would likely have to face substantial capital expenditures. When modeling applications that run in the cloud, application architects must consider how an application fits with the existing cloud model.

Infrastructure components like web servers, application servers, cache servers, and databases also need to be able to scale quickly when needed. If an organization uses large centralized components that are difficult to scale, these components may have to be re-architected to enable compatibility with the cloud – for example, large centralized databases may have to be de-normalized [31] and broken down into smaller segments. Further, configuration information and other data shared between cloud server instances must be easily accessible from newly-added components.

2.4 Cloud Deployment Models

There are three commonly-used cloud deployment models: private, public, and hybrid. An additional model is the community cloud, which is less-commonly used.

A **private** cloud is built and managed within a single organization. Organizations use software that enables cloud functionality, such as VMWare,vCloud Director, or OpenStack.

A **public** cloud is a set of computing resources provided by third-party organizations. The most popular public clouds include Amazon Web Services, Google AppEngine, and Microsoft Azure.

A **hybrid** cloud is a mix of computing resources provided by both private and public clouds.

A **community** cloud shares computing resources across several organizations, and can be managed by either organizational IT resources or third-party providers.

2.4.1 Managed Hosting

Some companies outsource their IT infrastructure management to third-party providers. In this case, services provided by the third-party provider may or may not be cloud-enabled. Typically these types of deployment models are similar to private clouds, with the only difference that these are hosted or managed by another company. There are also service providers who offer managed hosting services in public clouds, like Amazon and Google.

2.5 Cloud Computing Service Models

Cloud service models describe how cloud services are made available to clients. Most fundamental service models include a combination of IaaS (infrastructure as a service), PaaS (platform as a service), and SaaS (software as a service). These service models may have synergies between each other and be interdependent – for example, PaaS is dependent on IaaS because application platforms require physical infrastructure (see Figure 6).

The **IaaS (Infrastructure as a Service)** model provides infrastructure components to clients. Components may include virtual machines, storage, networks, firewalls, load balancers, and so on. With IaaS, clients have direct access to the lowest-level software in the stack – that is, to the operating system on virtual machines, or to the management dashboard of a firewall or load balancer. Amazon Web Services is one of largest IaaS providers.

The **PaaS (Platform as a Service)** model delivers a pre-built application platform to the client; clients needn't spend time building underlying infrastructure for their applications. On the backend, PaaS automatically scales and provisions required infrastructure components depending on application requirements. Typically, PaaS solutions provide an API that includes a set of functions for programmatic platform management and solution development. Google AppEngine is a popular PaaS provider, and Amazon Web Services also provides some PaaS solutions in addition to IaaS offerings.

SaaS (Software as a Service) provides ready online software solutions. The SaaS software provider has complete control of application software. SaaS application examples include online mail, project-management systems, CRMs, and social media platforms.

The main difference between SaaS and PaaS is that PaaS normally represents a platform for application development, while SaaS provides online applications that are already developed.

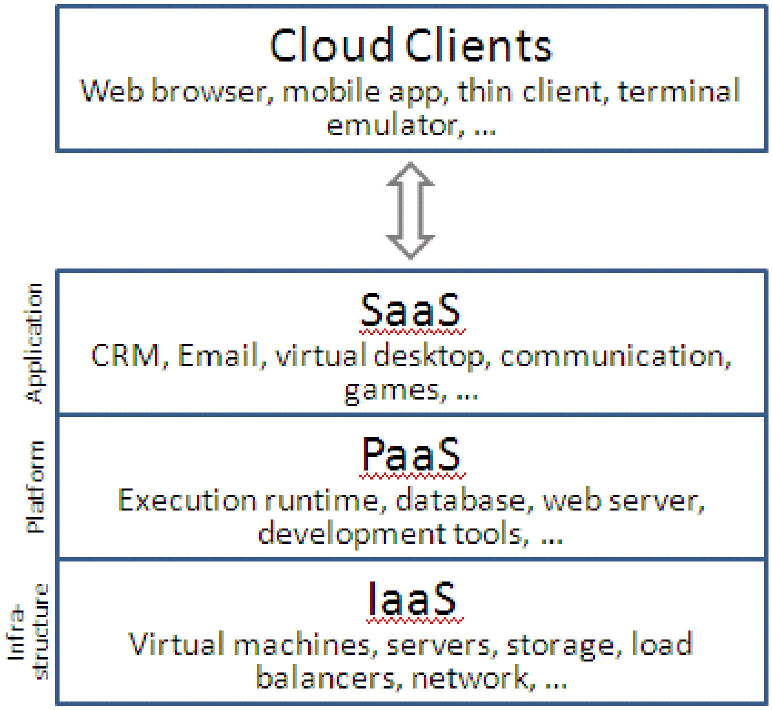


Figure 6: Cloud computing service models [5]

2.6 Nested Clouds

Several companies have become market leaders in the area of public cloud services. Cloud services such as Amazon (AWS) or Google AppEngine are the de-facto standards for cloud hosting. These providers went well beyond simply building the largest public clouds in the world; they also succeeded in defining the gigantic cloud eco-systems that have become platforms for other enterprise clouds.

Interestingly, many companies have built their own clouds within major public clouds. These organizations decided that building their cloud within a third-party cloud provides more benefits relative to building their own. One such company is Acquia.

Acquia is a leading provider of online products and services to help companies build and manage their websites based on the popular Drupal open-source social publishing platform. Acquia also offers a cloud hosting platform that helps companies host their websites. Acquia uses Amazon

Web Services to host both its own infrastructure and its customers' clouds. In February of 2011, the company had approximately 350 servers running in the AWS cloud [7]. Acquia CTO Dries Buytaert has described this decision: "Acquia chose AWS because it was the fastest way to get a new hosting service to market. It also saves us the cost of adding staff specialists on networking and infrastructure build out. Customers love our ability to quickly scale their sites using the elastic scalability of AWS and to quickly create clone sites for load testing" [7].

Beyond Acquia, public clouds have become launch-pad platforms for many small companies and start-ups. The ability to quickly take products to market without major up-front infrastructure investment provides substantial benefits. Given today's infrastructure and human-resources costs, early-stage technology companies are not able to start and sustain their business with traditional in-house IT infrastructure.

Even for cloud service providers like Acquia, the ability to "pay as you grow," to rapidly scale infrastructure, and to quickly take products to market outweighs potential long-term cost-saving benefits that a company could realize with their own cloud infrastructure.

3 CLOUD ADOPTION AND CONTROL CHALLENGES

Cloud computing has created a fundamental shift in how information technology infrastructure is run and managed, changing both the business and technology sides of IT. But, as with any major change in history, there are supporters and skeptics.

Transferring enterprise IT to the cloud is a complex task that includes both technical and organizational challenges. The cloud is a new paradigm that doesn’t have a clear one-sentence definition; it includes multiple factors, and therefore transformation to a cloud-based process may seem confusing. This complexity paired with uncertainty creates a number of organizational cloud-adoption barriers.

According to a recent survey (see Figure 7 [6]), security, cost uncertainty, and loss of control are the top three cloud-adoption barriers.

Figure 7: Cloud adoption barriers [6]

Those barriers represent business, technical and organizational challenges. The importance of organizational challenges is difficult to quantify, but such challenges can be critical in the decision-making process. Cloud advocates must be prepared to face substantial resistance, as employees inside organizations might be reluctant to embrace cloud-transformation and make this crucial technology shift.

3.1 Cloud Adoption Barriers

3.1.1 Data Security

Data security is by far the most challenging barrier to cloud adoption. Data is the most precious corporate asset, and companies want to know that their data is safe. Companies feel confident when they store data internally because they have full control over it. Although there is no guaranty that data is better protected internally comparing to public cloud. In fact, there is a possibility that data could be even safer in the public cloud because public cloud providers may poses higher level of data security expertise comparing to their customers.

When stored at public cloud, data can be compromised at several different data-lifecycle stages: during transfer from the internal company network to the public cloud, when data is stored in the public cloud, and during data backup and restore processes. There are fundamental questions to ask in order to ensure data security in a public cloud:

- Who has access to the data? What are the access-control policies? Do I have full visibility into information regarding these access-control policies?
- Is data encrypted during transfer from the internal network to the public cloud? What is the encryption algorithm? Can data be encrypted when stored in the cloud? Who holds the encryption keys?

  If a cloud provider is not supposed to have access to the data, encryption keys should be held only by the company that owns the data. Some of the compliance standards mandate full data encryption, and do not permit cloud providers to hold encryption keys.
- What is the disaster-recovery process? Does the cloud provider replicate data across multiple datacenters? Are these datacenters located in different geographical locations?

  If data is stored in only one datacenter and the cloud provider doesn't have the capability to replicate it at other datacenters, this should raise a red flag.
- What is the data-backup process? Who has access to the backup data? Where is the backup data stored?
- What is the data-recovery process? How long does data recovery take?
- What is the security-breach investigation process? Does the cloud provider have security-breach investigation capabilities?

  This question is often forgotten, but it is very important – if data is compromised, the cloud provider will be the only source of information for any investigation.

3.1.2 Cost Uncertainty

There are hundreds of different cloud offerings on the market, and their pricing models vary considerably. This creates uncertainty, and makes it difficult to estimate the real financial benefits of cloud computing. This uncertainty is particularly troublesome for non-technical people within organizations, who could be confused and scared away by the complexity of pricing models.

Pricing differences across cloud providers may include charges for data storage, CPU, memory, IP address, domain name service, and load balancing, among others. There is no single standard for cloud services pricing, and it can be complicated to compare prices between providers head-to-head.

It is also difficult to predict future prices with specific provider – and, once company is locked in with one cloud provider, it is not easy to move an entire infrastructure to other clouds.

3.1.3 Loss of Control

Loss of control factors can be subdivided into two types: technical loss of control, and organizational loss of control.

Technical loss of control includes such factors as access control, software versions and updates, and control over the timing of technical operations such as data backups and restores, and is partially related to the data security issues discussed above. To address technical loss of control challenges, a cloud provider should offer a company tools that provide full visibility into all cloud operations related to the company's assets.

Organizational loss of control is instead related mostly to human factors that may create barriers for the transformation to cloud computing. These factors may include the fears of some individuals of losing influence in the organization, the fear of job loss if the cloud transformation will functionally affect specific positions, and the simple inability to embrace change. The fear of losing control is common, and applies not only specifically to the cloud computing shift but also to many other organizational changes as well.

Another problem is that some organizations are just not ready to accept changes, or are very slow to implement these changes. Therefore, it is important to ensure the support of top management in order to implement change. Organizational loss of control is a complex issue, and is difficult to measure because organizations are typically reluctant to disclose these types of management problems.

3.1.4 Regulatory Compliance

Depending on business requirements, a company's computing infrastructure could be subject to certain compliance regulations. Companies should have a clear list of compliance requirements before considering cloud service providers.

Compliance regulations may vary. Some could be location-related, others security-related, and so on. The European Union Data Protection Act, for example, requires that all customer data about European customers be kept in European data centers. HIPAA compliance requires that all data be encrypted during both transfer and storage. Under HIPAA regulations, encryption keys should be accessible only to the company that owns the data, and not to the cloud provider.

3.1.5 SLA Agreements

An SLA (Service-Level Agreement)[32] agreement is a contract that describes the level of services offered by a cloud provider. In the case of cloud services, SLA could be measured in terms of mean time between failures, mean time to repair the outage and other operational metrics such as network response time and system performance.

Companies should perform due diligence to carefully examine a cloud provider's SLA agreements. Not every cloud provider wants to (or can) offer the level of business continuity required by organizations. Even cloud providers as large as Amazon provide only 99.95% guaranteed annual uptime for of their servers, while some organizations require 99.99% annual uptime. If service uptime drops below 99.95%, per Amazon's agreement customers are eligible for a service credit equal to 10% of their bill. Note that Amazon's SLA doesn't limit the length of downtime – whether your servers are down for two hours or 10 days, your company still receives the same compensation amount.

Interdepartmental service between IT and other departments inside a company are typically defined by operational level agreements (OLA)[33]. An OLA describes support responsibilities between each of these internal groups.

3.1.6 Data Portability/Integration

It can be technically difficult to integrate data in a company's internal datacenter with data located in a public off-premises cloud. Organizations that consider using a hybrid cloud where data is spread across both private and public clouds may face data integration problems:

- Security issues (data governance, network connectivity, etc.)
- Problems with transaction integrity (inability to support transactions across clouds)
- Difficulties handling large data volumes
- Lack of mechanisms to detect changes to data
- Data quality control issues
- Problems determining the origins of data

3.1.7 Software Compatibility

Cloud providers typically support a specific set of software vendors and versions. A public cloud is a shared environment, where software is shared among hundreds or thousands of isolated customer environments. It is critical for the cloud provider to maintain well-defined software standards, and therefore in many cases cloud providers cannot offer custom software packages installed to customer clouds. Particularly for PaaS or SaaS clouds, level of control over software is very limited. Companies must ensure that software in a public cloud is compatible with what they use internally.

3.1.8 Performance

Most cloud provider SLA agreements cover only infrastructure availability, not performance. If company applications have specific performance-related requirements, the company should discuss these requirements with cloud providers and confirm that these requirements can be supported. It is a good idea to include these requirements in an SLA contract, and it is standard practice to negotiate the SLA contact with the cloud service provider.

It is the responsibility of the cloud customer to monitor cloud performance and ensure that it is compliant with requirements and SLAs – collected performance metrics should be continuously analyzed. If cloud-hosted applications are used globally, it is important to monitor performance parameters such as network latency across all major customer locations.

When choosing a cloud provider, companies must estimate whether that provider will be able to support expected growth, and to guarantee sufficient performance levels as infrastructure scales up.

3.1.9 Lock-In Challenges

Each cloud provider offers a unique set of services and tools for operating and controlling its cloud. Learning a new cloud environments is similar to learning a new technology: it takes time and effort to master. This can be easier with IaaS-based cloud services, to which companies can install their own software on a provided infrastructure platform, but with PaaS or SaaS cloud platform customers must learn the provider’s specific interfaces and APIs in order to interact and manage these platforms. There are no uniform cloud management software standards across different providers, and once a company invests time and resources to establish operations on one cloud platform, it can be difficult to switch.

It is not recommended that a company become locked-in with a specific cloud provider, because this dependence may limit control over costs and technical cloud hosting options. It is not uncommon for cloud providers to raise service prices or alter their technology offerings.

How to avoid lock-ins?

There is no single answer to that as it depends on given use case and cloud-specific factors. These are generic guidelines, which may help to avoid lock-ins:

- Abstraction: Create or use existing abstraction layer in front of cloud provider. Cloud management services like Righscale or Scalr support multiple cloud vendors and allow to distribute cloud infrastructure across different providers. These services provide level of abstraction and may reduce dependencies on cloud providers.
- Avoid using customized vendor-specific services
- Make sure that cloud provider offers tools to migrate data off their cloud

4 TAXONOMY OF CLOUD SERVICES

This chapter presents a systematic study of cloud service offerings. As discussed in previous chapters, cloud service models can be divided into three main categories: SaaS (software as a service), PaaS (platform as a service), and IaaS (infrastructure as a service). Any of these service models can be implemented in private, public, or hybrid cloud environments.

4.1 Cloud Adoption and Service Offerings

Cloud computing started the largest IT transformation in history, and this transformation has opened many new business opportunities. It is expected that public clouds will provide most of the opportunities for cloud service providers. According to a recent survey conducted by Morgan Stanley [6], the percentage of companies using a public cloud is expected to rise to 51% though 2013. Three-hundred IT decision-makers were interviewed for this survey, and the responses clearly identify that public cloud adoption is about to reach critical mass. It is estimated that servers shipping to a public cloud will grow at 60% CAGR (Compound Growth Annual Rate), while on-site server spending will be reduced by 8.6% over the next two years. The survey suggests that public cloud computational workloads will grow 50% CAGR through 2013, compared to 8% CAGR for on-premise computational workloads (see Figure 7).

This quick public cloud adoption can be explained by several factors:

- No up-front investments or commitments are required.
- You pay for only what you use.
- Companies can test any services prior to purchase.
- Less human resources are required for infrastructure maintenance.
- Service offerings are easy to compare.
- Software upgrades can be automated.

Migration to the public cloud is expected to accelerate. While other deployment models – such as managed hosting, hybrid hosting, and private clouds – are expected to grow as well, public clouds will undoubtedly be the area of the most substantial growth. Companies that are better positioned to provide public cloud services are likely to benefit the most; therefore, most new cloud service offerings target public clouds.

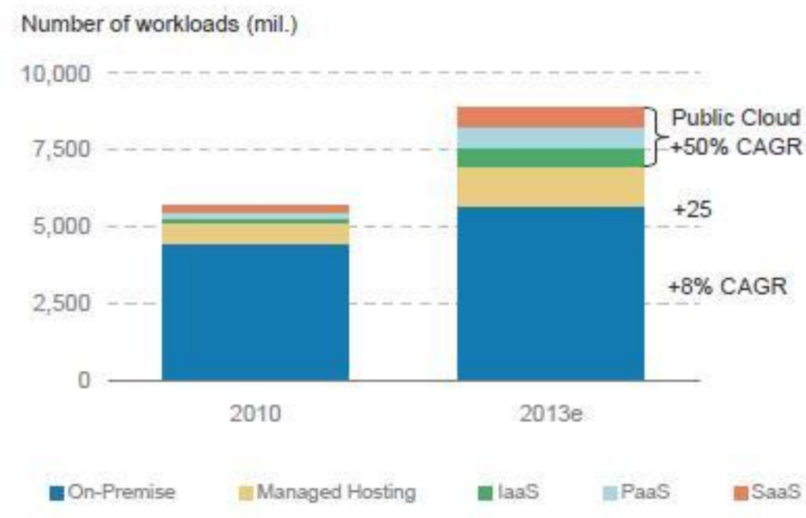


Figure 7: Expected cloud workloads [6]

Here, we describe and analyze cloud service taxonomy. A few companies are selectively reviewed from each taxonomy branch, and we focus particularly on public cloud taxonomies because that's where most cloud service offerings are. The goal of this cloud taxonomy analysis is to classify market offerings and to compare different pricing models across multiple providers.

Since all cloud service offerings can be classified into one or more cloud services models, we use the three fundamental models (SaaS, PaaS, and Iaas) as the top levels of taxonomy. Some might argue that many additional models could be identified, such as "database as a service" or "security as a service," and such arguments are valid because no official cloud services classification exists. However, we believe that all current models can be usefully classified under the three original ones, so this is the approach we take in this paper.

4.1.1 Public Cloud Services Taxonomy

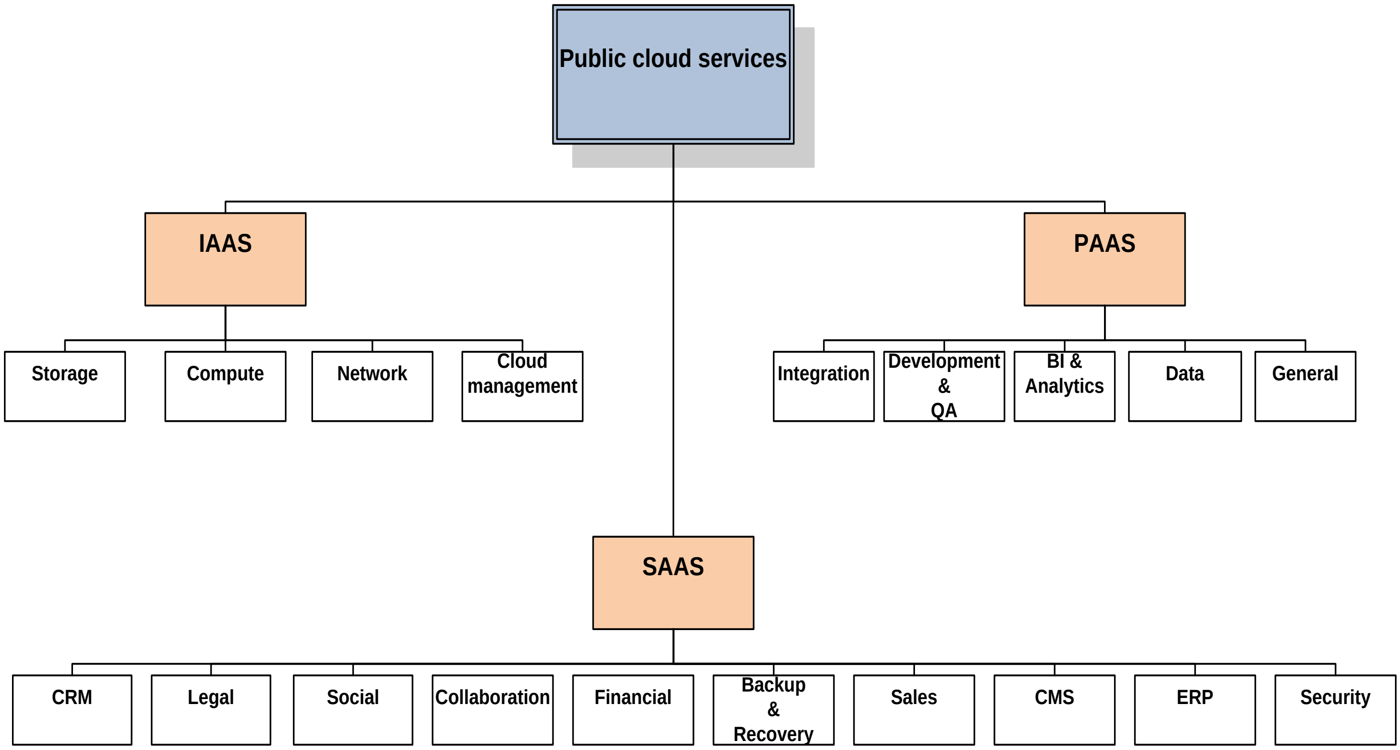


Figure 8: Public cloud services taxonomy, top level

4.2 IaaS Services

Infrastructure as a service helps companies to move their physical infrastructure to the cloud with a level of control similar to what they would have in a traditional on-premise datacenter. IaaS provides the closest resemblance to the in-house datacenter compared to other services types. Core datacenter infrastructure components are storage, servers (computing units), the network itself, and management tools for infrastructure maintenance and monitoring. Each of these components has created a separate market niche.

While some small companies specialize in only one of these IaaS cloud niches, large cloud providers like Amazon or RightScale have offerings across all IaaS areas. Technically, the IaaS market has a relatively low barrier of entry, but it may require substantial financial investment in order to build and support the cloud infrastructure. Mature open-source cloud management frameworks like OpenStack are available to everyone, and provide strong a software foundation for companies that want to build their private cloud or become a public cloud provider. Hewlett-

Packard recently announced plans to offer public cloud services, and plan to build their cloud using OpenStack. Several other cloud providers, including Rackspace, have been running their services on OpenStack for a some time.

Figure 9 lists the most important market players in each of the IAAS areas.

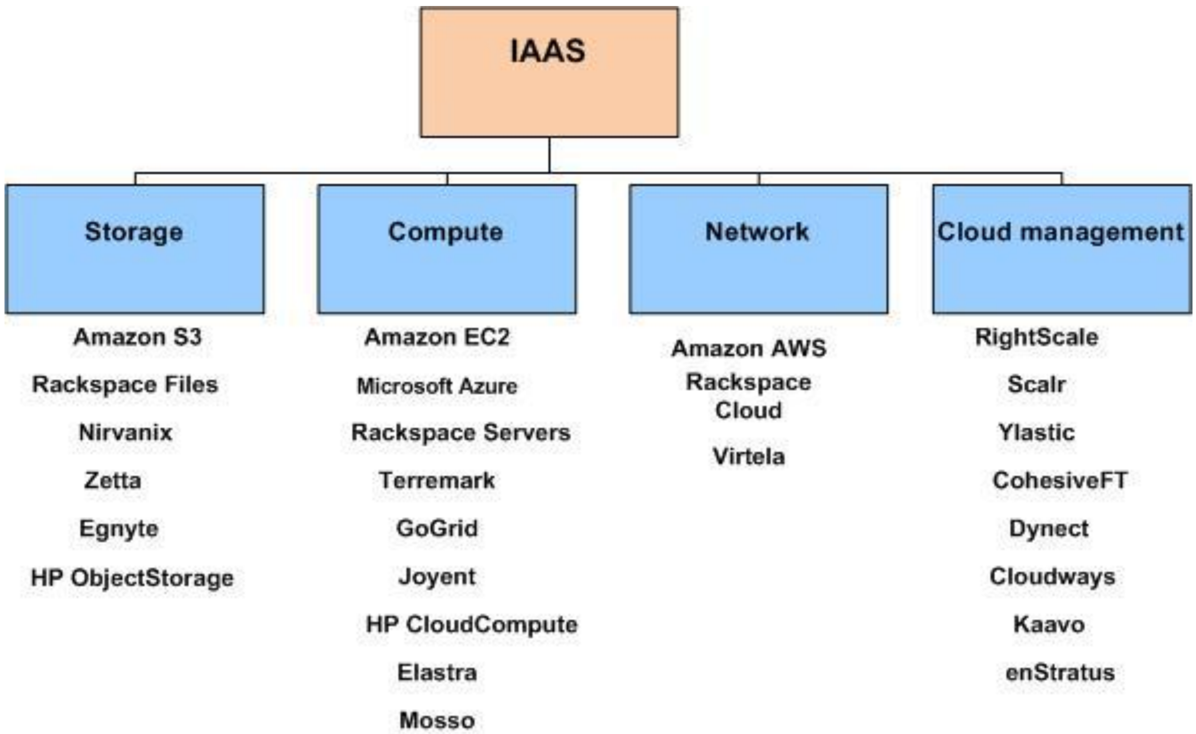


Figure 9: Public cloud services taxonomy, IaaS

4.2.1 IaaS: Storage

Pricing: $/size of stored data, $/amount of incoming data traffic, $/amount of outgoing data traffic, $/number of specific http requests **Pricing options may vary among different providers*

Storage services allow companies to store data on the storage devices of third-party providers. Cloud storage is accessible online and presented to customers as a set of storage pools or buckets, accessible using rich interfaces such as programming APIs, web interfaces, or command-line tools. The complexity of cloud storage architecture is hidden from clients, but on the back-end it is quite complex – typically composed of distributed storage devices that are managed by centralized software. Sophisticated storage management software employs algorithms that manage data distributed across multiple storage devices.

However, cloud storage may not fit every organization's needs. Potential disadvantages include network latency, dependency on internet availability, security considerations, and limited control. Network latency is higher compared to in-house storage because the cloud provider's datacenter is located in a different geographical location at least few network hops from the client's location. A client who stores all data in a public cloud and doesn't have a local copy is totally dependent on internet connectivity. A cloud provider should offer high-level security to avoid information loss or compromise, and data transfer must be encrypted.

Characteristics of typical cloud storage:

- Highly reliable and redundant
- Automatically scalable
- Self-service provisioning available for customers
- Accessible through rich interfaces (Web console, API, CLI)
- Pay-as-you-go payment model

4.2.2 Cloud Storage Pricing

Most cloud storage providers determine pricing using the following factors:

- Total amount of stored data
- Amount of data transferred to storage cloud
- Amount of data transferred out of storage cloud
- Number of specific http requests

As an example, let's compare pricing between Amazon S3 and Rackspace Cloud Files storage.

	Amazon S3	**Rackspace CloudFiles**
Storage	0.11/GB/month	0.10/GB/month
Data Transfer IN	Free	Free
Data Transfer OUT	0.09/GB/month	0.18/GB/month
HTTP requests (PUT, COPY, POST)	0.01/1000 requests	Free
HTTP requests (GET, HEAD)	0.01/10000 requests	Free

Figure 9: Amazon S3 vs. Rackspace CloudFiles pricing comparison (* 40 TB storage)

Based on the comparison in Figure 9, Rackspace storage would likely be more expensive for a company that serves many large files, like images or videos; however, Amazon could be more expensive for a company that serves many small files, but whose site traffic is very intense.

		Amazon S3	Rackspace CloudFiles
Storage	40TB	4415	4000
Data Transfer IN	5TB	0	
Data Transfer OUT	25TB	2557	4500
HTTP requests (PUT, COPY, POST)	1M	10	0
HTTP requests (GET, HEAD)	100M	100	0
Total:		**$7082/month**	**$8500/month**

Figure 10: Amazon S3 vs. Rackspace CloudFiles monthly cost ($) comparison (*40 TB storage in U.S. East datacenter)

4.2.3 IaaS: Computing

Pricing:

On-Demand	Prepaid (reserved)	Spot (auctioned)
$/hours of service	$/year + $/hours of services	$/hours of service

**Pricing options may vary among different providers*

Computing services provide computational resources for customers. These services include CPU, random access memory (RAM), and I/O resources. Computing resource pricing options may vary between different providers, but generally pricing options are determined by the amount of computing resources and by overall payment models. Computing resources are offered as virtual machine instances, whose instance types and assigned prices depend on the combination of CPU, RAM, and I/O capacity. Providers offer a few types of instances, which cover most customer needs and make it easy for customer to choose (i.e., small, medium, large, etc…).

The most popular pricing models are on-demand, prepaid (reserved), and auctioned.

4.2.3.1 On-Demand

These types of instances allow customers to pay hourly service fees without long-term commitments. This pay-as-you-go pricing model is ideal for situations when a company cannot estimate computing resources demand up-front.

4.2.3.2 Prepaid

With prepaid services, customers pay a fixed amount up-front for a specific commitment period. Normally, you pay lower costs for longer commitment periods because this helps cloud providers to estimate their infrastructure expenses.

4.2.3.3 Auctioned

The auctioned pricing model enables customers to bid for the computing capacity offered by a cloud provider, with bid price regulated by supply and demand. If your bid meets or exceeds the current bid price, you can access the resources. If your bid is overridden, you give resources back. The costs for auctioned computing resources are significantly lower compared to prepaid and on-demand services; however these resources cannot be used for critical production environments because they can be taken away if a customer loses a bid.

4.2.4 Which Pricing Model to Choose?

Choice of pricing model depends on specific customer requirements. The on-demand model is the right choice for customers who cannot estimate their computing resource needs. Popular uses for on-demand computing include overflow capacity for companies that host their own infrastructure and use Amazon only when additional capacity is required. This model also fits well with start-ups and smaller companies with limited funds, because they can pay for services as they grow. The prepaid model provides cost benefits when a company can estimate their long-term demand for computational resources. This model works particularly well for larger organizations that can estimate required capacity and can afford to make an initial investment. Auctioned resources are typically used for non-production environments, when customers can afford to lose their virtual machine for a short period of time. Technically, auctioned instances can be used in mission-critical environments as well, but in such cases companies should place high bids to ensure uninterrupted service and availability.

Prices for cloud computing resources constantly decrease across multiple providers. As more providers enter the market, competition forces providers to offer a wider range of price options in order to reach more customers.

	On-Demand	**Prepaid (reserved)**	**Spot (auctioned)**
August, 2012	$0.080/hr	$160/year + $0.024/hr	$0.007/hr
March, 2011	$0.085/hr	$227.50/year + 0.03/hr	$0.031/hr
February, 2009	$0.10/hr	N/A	N/A

Figure 11: History of Amazon EC2 computing costs (based on a "small" Linux EC2 instance – 1.7GB RAM, 1CPU, 160GB storage – in the U.S. East region)

4.2.4.1 *On-Demand vs. Prepaid*

Here, we compare Amazon EC2 on-demand against prepaid instance costs, based on a one-year timeframe. The cloud server instance used in this comparison is a "small" type (1.7GB RAM, 1CPU, 160GB local storage) Linux instance running in a U.S. region.

Cost calculation is based on following costs:

On-Demand	**Prepaid (reserved)**
$0.080/hr	$160/year + $0.024/hr

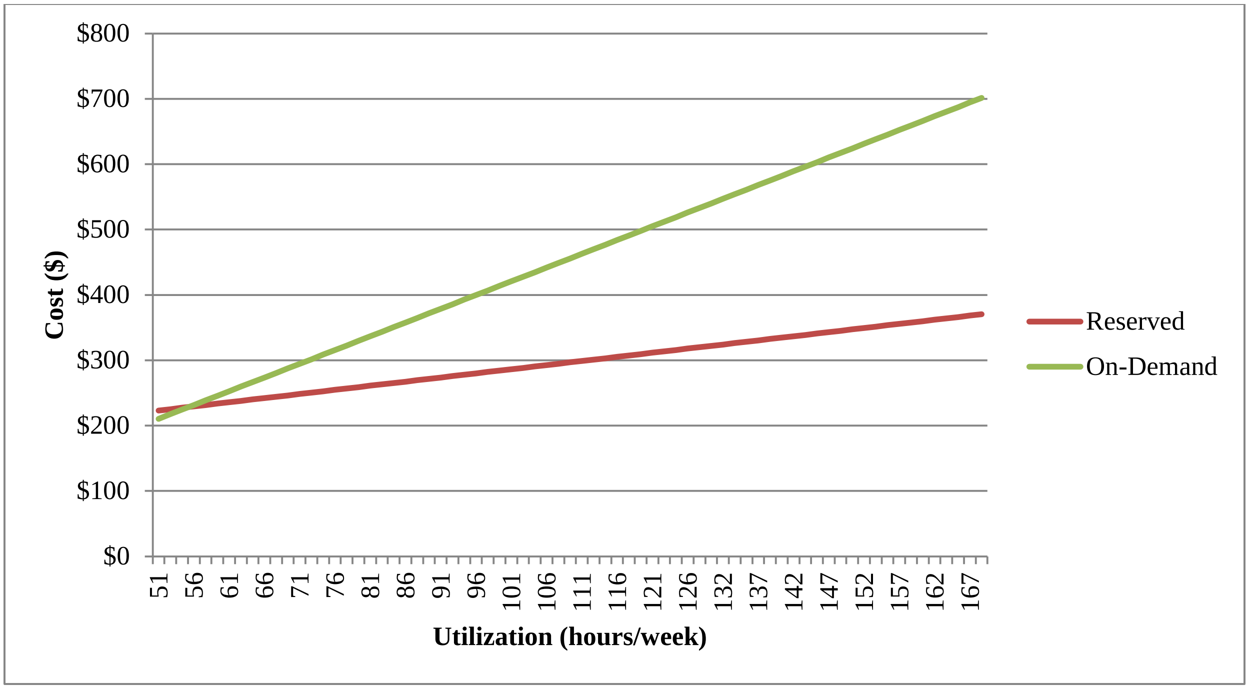


Figure 12: Amazon EC2 cloud instance pricing comparison, reserved vs. on-demand (based on a "small" Linux EC2 instance – 1.7GB RAM, 1CPU, 160GB storage – in the U.S. East region)

Figure 12 shows EC2 on-demand and reserved (prepaid) server instance yearly costs relative to utilization. Server utilization is measured by how many hours the server is used per week. Technically, if a server is started then that server's resources are being used, so customers are charged the same price whether CPU/RAM utilization is very low or extremely high.

The break-even point between these two pricing types is around 56 hours per week, 33% of maximum instance utilization. As such, if a company expects their server utilization to be more than 33%, and they expect to use Amazon services for at least one year, financially it makes sense to go with a reserved instance

Utilization			Costs	
Hours per Week	Hours per Year	Utilization %	Reserved	On-Demand
56	2,893	0.33	$229	$231
100	5,172	0.59	$284	$414
168	8,766	1	$370	$701

Figure 13: Amazon EC2 cloud instance pricing comparison, reserved vs. on-demand (based on a “small” Linux EC2 instance – 1.7GB RAM, 1CPU, 160GB storage – in the U.S. East region)

Clearly, Amazon offers more attractive prices for reserved prepaid instances compared to on-demand. In addition, historical cost reductions were more aggressive for prepaid instances (see Figure 11). In their last price reduction, Amazon dropped their on-demand instance prices by up to 10%, and reserved instance prices by up to 37%.

There are several reasons that cloud providers offer more aggressive prices for prepaid computing resources:

1. Providers want to lock customers into their cloud, and so prevent them from using the services of other providers.
2. Providers can better plan their cloud capacity for the years ahead. The ability to forecast capacity helps providers to make correct capital expenditure (CapEx) decisions.
3. There are low barriers to enter the IaaS market, and new entrants periodically disrupt the market by offering new technologies and more competitive processes. This creates a challenge for established cloud providers, and makes it more difficult to keep customers. Prepaid services ensure long-term customer commitment and minimize defection to upstart services.

4.2.5 IaaS: Network

There are two major network services offered by public cloud providers: load balancing and DNS (domain name systems). Detailed technical descriptions of these services are beyond the scope of this paper, but we provide a brief introduction to these technologies in order to set the context for further analysis.

4.2.5.1 Load balancing

Pricing: $/hours of service (only when service is used)/month + $/amount of consumed bandwidth (GB) /month **Pricing options may vary among different providers*

Load balancing provides a single point of access to multiple servers that run behind it. A load balancer is a network device that distributes network traffic among servers using specific load-balancing algorithms. Many different load-balancing algorithms exist, although the most popular include the following:

Round-robin: even connection distribution across all servers

Weighted round-robin: connection distribution proportionate to the weight assigned for each server

Dynamic round-robin: similar to weighted round-robin, but server weight is dynamically determined based on continuous server monitoring

Least connections: connection is sent to the server with the lowest number of current connections

Fastest: distributes new connections to a server based on the fastest server-response time

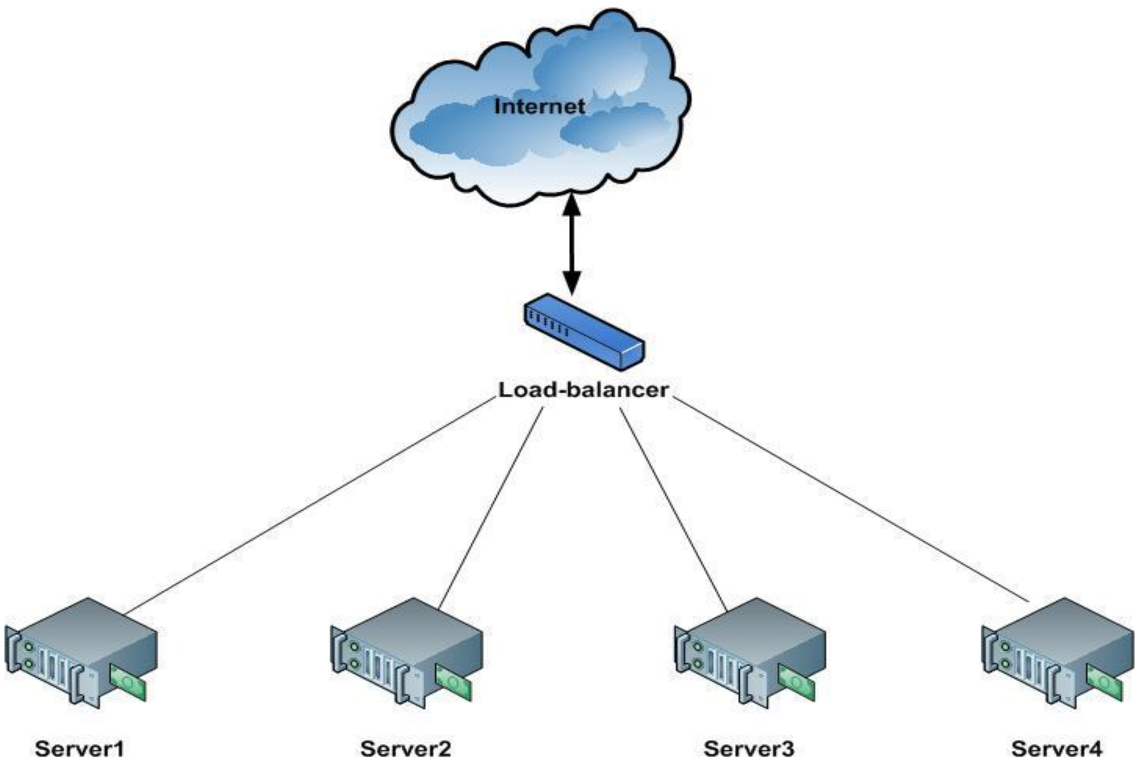


Figure 14: Load balancing

There are several benefits from load balancing: **failover** – in case of specific server failure, the load balancer will automatically forward network traffic to other servers; **performance** – because traffic load is distributed between multiple servers, network response time is typically faster; **scalability –** customers can quickly add servers under the load balancer to increase computational capacity without affecting other network/system components.

Load-balancing cloud services provide load distribution only for internal instances deployed within the provider's cloud, with service price determined by hours of service plus amount of consumed bandwidth. Different load-balancing service providers may offer different technical features – see Figure 15**,** where we compare Amazon and Rackspace load balancers.

Feature	Amazon	Rackspace
Public IP address[33]	available	available
Access control list[34]	not available	available
Connection logging	not available	available

SSL termination[35]	available	not available
Connection throttling[39]	not available	available
LB algorithms[36]	round-robin only	round-robin, random, weighted rr, least conn.
Internal virtual IPs[37]	not available (only in VPC)	available
IPv6 support[38]	available	available

Figure 15: Amazon vs. Rackspace load-balancing services comparison

4.2.5.2 Domain Name System (DNS)

Pricing: $/hosted DNS zone/month + $/number DNS queries/month

**Pricing options may vary among different providers*

DNS is a hierarchical naming system for computers, or any other naming devices that use IP addressing for network identification – a DNS system associates domain names with IP addresses. A cloud provider's DNS services offer to store IP/domain-name mapping for both internal and external servers. Pricing models may differ between providers; some (like Amazon) charge per DNS-hosted zone and number of DNS lookups, while others (like Rackspace) offer DNS services for free to their existing customers.

There are a few differences between the DNS features offered by cloud providers and in-house DNS services:

- Cloud providers offer a DNS management API for developers. These APIs allow developers to automate DNS operations such as authentication and domain-zone creation
- High-availability and automatic scalability
- Extended security and access control
- Geolocation-based DNS routing

4.2.6 IaaS: Cloud Management

Pricing: $/ number of managed servers + service features + level of support

**Pricing options may vary among different providers*

Cloud management service providers are niche-market players who help customers to simplify cloud management. Cloud management services (CMS) tend to fill some of the cloud management tool gaps of large cloud providers. Major cloud providers concentrate on the core of their business by providing scalable and reliable cloud infrastructure to customers, but in some cases providers may lack the resources (or inclination) to develop complementary cloud management services. Large cloud providers typically take a “good enough” approach for complementary cloud management tools because these tools don’t directly contribute to revenue. CMS providers work directly with specific cloud providers to integrate provider services with a customer’s existing cloud, or to create value added services on top of the provider’s existing infrastructure.

Customers benefits from the additional capabilities provided by cloud management services, which may include integration services, security and access control services, high-availability features, and database-replication services. CMS can be classified into three main branches: **cloud value added management services, cloud integration services**, and **cloud service brokerages**.

4.2.6.1 Cloud Value Added Management Services

Cloud value added management service (CMS) providers enhance existing cloud provider services by building additional features on top of them. CMSs are typically used for public cloud services; however, some support private cloud solutions, such as VMWare or Citrix. CMSs bring value to customers because they save time and resources by automating regular tasks that customers would otherwise have to implement themselves.

Scalr is a popular CMS provider that works directly with the Amazon and Rackspace public clouds, and provides value added services such as database fault-tolerance and auto-scaling[40] , backups, security, DNS management, and distributed task scheduling.

In order to demonstrate the value provided by CMSs, we compare Scalr services with Amazon’s AWS services, and review AWS features enhanced by Scalr.

Feature	Amazon	Scalr (enhanced features)
Database (MySQL) scalability and high availability	Database master and slaves with automatic replication.	**Automatic DB master failure detection** and slave-to-master promotion **Database auto-scaling** creates additional database slaves based on increased load conditions **DB performance improvement** by splitting read and write requests across different nodes
EC2 virtual machine auto-scaling[40]	Auto-scaling available through command line tools	**Easy-to-use web UI tool** to specify auto-scaling policies **Flexible component-based auto-scaling policies** **Custom auto-scale metrics** **Smart downscale** event handling: graceful shutdown, downscale pre-condition triggers
Backups[43]	No offsite or inter-cloud backup features available	**Offsite backup** tools available **WebUI interface** for backup management
Scheduler[41]	Only standard host-based cron scheduler	Centralized cluster-based and host-based scheduler with simple web UI control.
Logical server grouping	Not available	**Logical server grouping** helps to holistically manage servers. Customers can operate with a group

		of servers instead of repeating the same operation on individual hosts
DNS management[42]	Web interface available for DNS records management	DNS management **policy based web UI** **Automatic DNS record addition/termination** upon computing instance creation/removal
Security	Web interface available for firewall management	Web interface for firewall management. Automatically modifies firewall to allow connections only from other servers in the same cloud.

Figure 16: Scalr features enhancement over AWS

Cloud value added management services make sense because they save time and resources for companies. The challenge for CMS providers is to stay ahead of the curve in making sure that services they provide are relevant. If a large cloud provider implements some of these value added services, then it could easily take the CMS provider out of business. Cloud providers like Amazon and Rackspace possess extensive cloud management expertise, and know how to manage their clouds; therefore, they can easily replicate any services built on their platform if they find that a given service can generate reasonable profit.

Another challenge is that there is a low barrier to enter the CMS business because CMS providers generally utilize existing technologies and APIs offered by cloud providers. CMS companies don't create much intellectual property of their own, and their software features are easy to replicate.

4.2.6.2 Cloud Integration Services

Cloud Integration Services (CIS) provide integration between private and public clouds. CIS providers offer a management abstraction layer that enables transparent multi-cloud support – customers can manage different clouds in both public and private environments using centralized administrative tools and APIs. A CIS can automate routine administrative tasks such as provisioning, scaling, and monitoring and configuration management with easy-to-use centralized administration tools.

CISs provide multi-cloud governance tools to allow customers to monitor all logs, security events, and computer resources utilization from a single dashboard. In addition, such centralized governance tools help to control user access, maintain server security, and perform security audits. Users can set conditional triggers to automatically send notification alerts in case a specific predefined condition is reached.

Administrative cloud-management tasks require time and resources, and although there are similarities each cloud-management technology is different. Learning new cloud-management tools is similar to mastering a new programming language: it takes effort to develop a significant level of expertise. When companies employ hybrid clouds or cloud technologies from different providers, this learning curve becomes even steeper because there are more technologies to learn. A major benefit of CIS is that they provide a transparent management abstraction layer that offers exactly the same controls to manage multiple clouds. This helps to minimize the learning curve, simplify management tasks, and reduce administrative overhead.

Rightscale is a major cloud-management services company that offer cloud integration services, one of the earliest cloud-management companies on the market in 2006. Rightscale customers include Associated Press, CBS, and Zynga. Currently, Rightscale provides management support for most of the largest public clouds across the world, such as Amazon Web Services, Rackspace, SoftLayer, Google Cloud, Windows Azure, Tata, and Korea Telecom. Rightscale also supports most popular private cloud technologies, including CloudStack, OpenStack, and Eucalyptus.

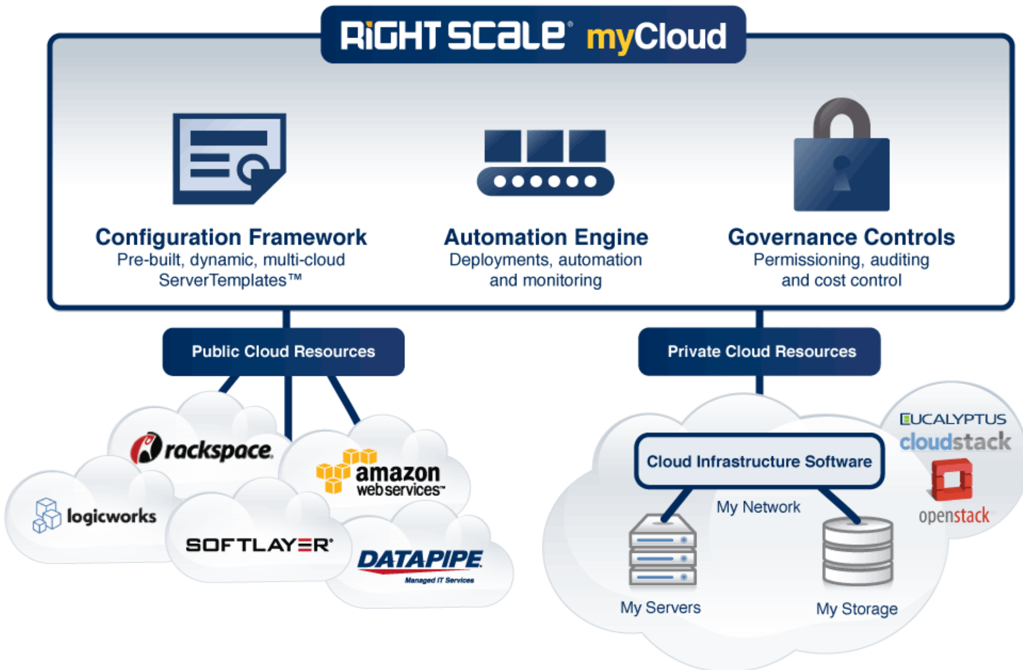


Figure 17: Rightscale private-public cloud integration [13]

Cloud integration service providers have lower long-term business risks compared to cloud value added management service providers, because they enable hybrid cloud services and provide integration between private and public clouds. The ability to provide hybrid cloud solutions and integrate multiple clouds minimizes dependencies on specific public cloud providers, and offers certain protections from provider technology overlap risks.

4.2.6.3 Cloud Service Brokerages

Cloud service brokerages (CSB) aggregate the services offered by multiple cloud providers and organize these services into a service catalog. This service catalog provides easily-searchable product information for the IaaS, SaaS, and PaaS services available for purchase though the CSB provider distribution network. CSB services can be viewed as a cloud services marketplace, where customers can quickly identify and purchase required services. In addition to service brokerage, these providers also enable centralized cloud services management in a manner similar to what CMS and CIS providers do. In addition, CSBs can help both service providers and their customers to manage administrative tasks like billing and arbitration.

The primary difference between cloud service brokerages, cloud managed services, and cloud integration services is that any cloud services provider can sell services through cloud brokerages, while CMS and CIS companies work only with providers they choose.

According to many analysts, CSB is forecasted to be one of the most dynamically-growing segments in the cloud-computing industry. "Cloud services brokerages makes it easier, safer and more productive for companies to navigate, integrate, consume, extend and maintain Cloud services, particularly when they span multiple, diverse Cloud services providers" – Gartner Group.

CSBs provide multiple benefits for both cloud providers and customers by enabling distribution channels for cloud providers, which could be particularly helpful for the smaller providers who don't have a long customer reach. Using such distribution channels, cloud providers can resell not only their own services but also related third-party solutions.

For consumers, CSBs can serve as a centralized access point to a wide range of certified cloud services – they offer a single place for services provisioning, policy enforcement, compliance, security, audit, billing, and other administrative tasks. CSBs can also help customers to integrate new services with existing infrastructure, and to establish full visibility into their cloud infrastructure though monitoring and reporting tools.

4.2.7 Interview with Sebastian Stadil, Scalr CEO and founder of Silicon Valley Cloud Computing Group

Hi Sebastian. Could you please give me some background about Scalr?

I first came up with this business idea around ten years ago when my web server crashed under heavy traffic and the hosting provider wasn't able to dynamically scale it. We started Scalr in 2009, and in 2010 we decided to open-source our platform code [Note: Scalr sells Scalr platform support services]. It makes sense to keep the code open to the community to make it easier for everyone to use and customize.

Why do people want to use Scalr?

People want to use Scalr because it helps them to run things smoothly. It helps to dramatically improve productivity and reduce costs.

Would you agree that Scalr's core business model is to complement large cloud providers like Amazon by filling their gaps and enhancing their product features?

That's how we thought before. But then Amazon was always catching up with us, and we realized that it makes more sense to develop our own product rather than filling gaps in existing solutions offered by major cloud providers. Developing our own product brings more value for customers and helps to minimize business risks.

How do you stay ahead of the market and protect yourself from being outplayed by competitors?

The best protection for us is the speed of development and quality of our product. We can develop and add new features much faster than larger companies. Talking about smaller competitors and new market entrants, it is going to be quite difficult for them to catch up with us because we are already few years ahead in development – we have a great team and lots of cloud computing expertise.

Do you always need to stay on the edge of technologies and respond to new market trends quickly? How do you prioritize development of new features?

We prioritize our development based on the requests from our largest customers. We feel that jumping on every new market trend just because it's cool and spending limited resources on it is the wrong approach.

How do you see the cloud computing market in the long-term?

I think we are going to see lots of consolidation. At the end there will be a few large players holding control over the market. Cloud services will be federated and standardized. I don't believe that niche-market players will survive; some of them will be acquired, others will be out of business.

From the cloud platform perspective, I think there will be a few large providers like Amazon, Google, and Microsoft which use proprietary cloud technologies, competing against providers like Rackspace which use open-source platforms, such as OpenStack.

Another trend happening now is that cloud computing IaaS resources like virtual machines and storage are getting commoditized. So, I believe that companies which are going to survive are likely to be the ones providing value services on top of the stack. I'm talking about PaaS, SaaS, and cloud-management services which work on top of IaaS.

What do you think is the future of the private cloud?

The private cloud will exist as a segmented market for specific use cases. Most of the computing will be done in the public cloud.

What do you think about new cloud ideas like RaaS [resource as a service]? RaaS assumes that cloud providers can sell computing resources in more granular way – that is, instead of selling whole VMs they can separately sell CPU and RAM.

Selling computing resource components separately would be very hard; it would require major technical changes. Cloud providers slice their servers into virtual machines. Virtual machines can be sized and priced based on the customer requirements. That way it is easy to measure resource utilization and determine costs.

4.3 PaaS Services

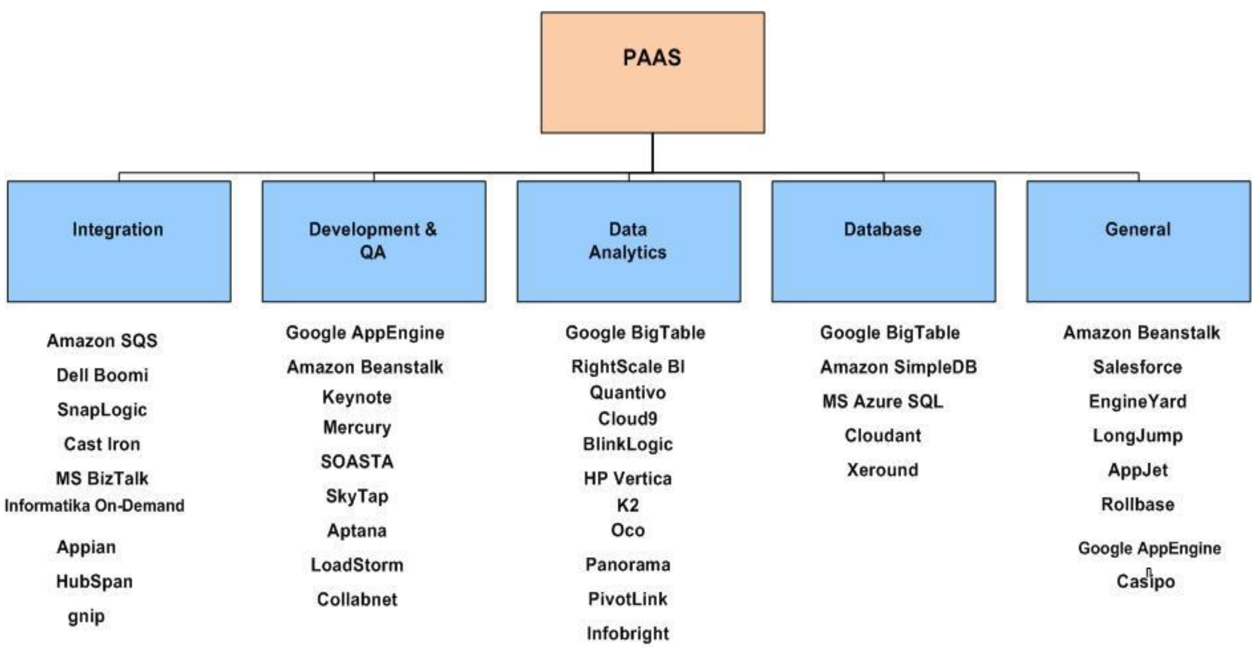


Figure 18: Public cloud services taxonomy, PaaS

PaaS (Platform as a Service) is a cloud service model that provides customers with a configurable application platform including a pre-installed software stack. PaaS can be viewed as another abstraction layer above the hardware, operating system, and virtualization stack (see

Figure 6). The PaaS model brings significant value to companies because it reduces complexity of infrastructure and application maintenance and allows concentrating on core software development competencies.

As mentioned in the introduction chapter, the cost of software development in large organizations is typically lower than the cost of software and infrastructure maintenance. Unsurprisingly, companies are increasingly interested in simplifying their application and middleware infrastructures in order to improve productivity and minimize related operational expenses.

4.3.1 PaaS Service Characteristics

- Scalability and auto-provisioning of the underlying infrastructure
- Security and redundancy
- Build and deployment tools for rapid application management and deployment
- Integration with other infrastructure components such as web services, databases, and LDAP
- Multi-tenancy, platform service that can be used by many concurrent users
- Logging, reporting, and code instrumentation
- Management interfaces and/or API

The PaaS market segment is less mature than IaaS – there are only a few large PaaS providers, along with multiple smaller-market niche players. Although PaaS and SaaS may require deeper technical expertise to develop compared to IaaS, they also don't require as much initial infrastructure investment. It is reasonably easy and affordable to build PaaS or SaaS infrastructure within the existing IaaS clouds of large providers like Amazon or Rackspace. This is the route typically followed by smaller companies entering the PaaS and SaaS market. The PaaS segment is still very much in development, and provides considerable potential opportunities for innovators. A multitude of application and middleware software services that are currently run internally will likely be taken to the public cloud in the next few years; accordingly, we can expect a certain level of market consolidation in this area:

> During the next two years, the fragmented, specialized PaaS offerings will begin to consolidate into suites of services targeting the prevailing use patterns for PaaS. Making use of such preintegrated, targeted suites will be a more attractive proposition than the burdensome traditional on-premises assembly of middleware capabilities in support of a project. By 2015, comprehensive PaaS suites will be designed to deliver a combination of most specialized forms of PaaS in one integrated offering. – Gartner Group, 2011

In the PaaS taxonomy (see Figure 18) we list the leading companies in the PaaS space. Five major branches of cloud platform services have been identified, and are selectively reviewed: integration, development and QA, data analytics, database, and "general" (providers that offer multiple PaaS services).

4.3.2 PaaS (SaaS): Data Analytics and Business Intelligence

Pricing: Varies by storage, number of queries, amount of processed data, user licenses, support options, and other factors.

Data analysis and business intelligence (BI)[45] software helps companies to simplify data operations and apply statistical analysis methods in order to support business decision-making and information retrieval.

Data analytics and business intelligence is considered one of the most untapped opportunities in the cloud. BI solutions are very costly, involve a high level of technical complexity, and require significant expertise to implement and maintain. Moreover, data is probably the most precious asset of any organization, and security always stays on top of other concerns. Not every company is willing to take risks and transfer data to the public cloud, even if the data is encrypted and the cloud provider is fully compliant. Security is one of the major reasons that cloud adoption was much slower in the data analytics and BI space compared to other IT segments. The benefits of a public cloud, however, are often too attractive to miss, and can outweigh even sensitive security concerns.

According to a recent survey [18], approximately one-third of organizations already use or plan to use business intelligence to complement or improve their internal capabilities. There are key advantages driving BI into the cloud.

4.3.2.1 Elasticity and Pay-as-you-Grow

As discussed earlier, elasticity is a major benefit of the cloud, and as with any other IT segment BI can strongly benefit from the elastics of cloud computing. Data analysis and statistical modeling tasks require massive computational capacity – these types of computation workloads are very difficult to predict, and therefore companies must always maintain extra hardware and overprovision their infrastructure up-front. Overprovisioning can be avoided by moving part or all of the analytics infrastructure to the public cloud. Some companies prefer a hybrid cloud approach in which they run their computations in an on-premise cloud, but use public clouds when additional capacity is required; others prefer to move their full analytics infrastructure stack to the cloud.

Public cloud computing has opened new data analytics BI opportunities, particularly for smaller companies and startups. The initial investment required to start BI operations internally on-premises is substantial, and not every company can afford such an investment. Public BI services, however, don't require up-front investment and instead offer pay-as-you-go services, which opens BI computing opportunities to everyone.

4.3.2.2 Cost Structure

Software service subscription in the cloud is considered an operational expense rather than capital. Operational expenses provide tax benefits to companies because these expenses can be deducted in the same tax year, as opposed to capital expenses that can be only partially deducted every year as they depreciate.

Some experts think that running analytical services infrastructure in the public cloud is cheaper compared to an in-house cloud, but this is still unproven. BI in the public cloud is likely to be cheaper in the short-term because no initial investment is required; however, long-term cost benefits are likely to come from other factors, such as reduced maintenance and support costs.

4.3.2.3 Lack of Skills and Internal Expertise

Demand for data analysts and statisticians is very high, and is expected to grow (see Figure 21). "Data scientist" is currently one of the hottest job titles in the market, the skills typically required for which include advanced statistical analysis and math. It is very expensive for companies to hire data scientists, and it takes a long time to develop internal expertise in this area. SaaS and PaaS platforms will not completely resolve skill-shortage issues, but they can certainly hide some complexity and simplify data-mining tasks.

Figure 21: "Data scientist" job listings trend from listing aggregator indeed.com

4.3.2.4 Data Integration

Cloud integration solutions like Dell Boomi (see PaaS: Integration) make it easier to integrate data with multiple external sources and share data with customers and partners. Cloud integration solutions also help to maintain data mappings by automatically propagating changes in specific data sources across integration clouds. Cloud data integration technologies enable small and medium-sized businesses to more quickly integrate their systems with external data sources. Large-scale data integration has always been challenging in on-premise infrastructure because of the high cost barriers and technical complexity.

4.3.2.5 Time to Value

PaaS/SaaS solutions enable faster deployment, prototyping, and testing of the product. Fast deployment and infrastructure management processes are typical not only for the BI solutions, but also for all other cloud-enabled technologies in general. In the highly-competitive technology business environment, it is critically important to respond quickly to market trends and customer demands.

4.3.2.6 Maintenance

Public cloud BI service providers resolve the high underlying complexity of BI infrastructure and maintenance by automating routine maintenance tasks such as software patches and updates. This helps companies to reduce IT support and maintenance costs.

4.3.3 Public Cloud BI Advantage Comes at a Cost

However, there are also disadvantages and costs to cloud business intelligence models.

4.3.3.1 Control

As mentioned in the "Cloud Adoption and Control Challenges" section, companies may not have full visibility into public cloud operations (storage, backup, network transfer) and security (access control, encryption) related to their data. Granularity of data governance depends on the features offered by the cloud provider.

4.3.3.2 Customization

Public cloud services may have thousands of customers, and so such services cannot fully satisfy every customer need. Providers typically offer "good-enough" services and sets of features that are appreciated by a majority of customers. If additional features are required, there is no guarantee that the provider can add them.

4.3.3.3 Maturity

Many public cloud BI solutions are offered and built by relatively young companies and start-ups. So, it is not surprising that their solutions may not provide the same rich set of features found in on-premise solutions from companies like Oracle, Teradata, or SAP. Public cloud BI services are maturing, but it is will take some time for them to catch up with established technologies. Companies planning to migrate their BI technologies to the public cloud must do their diligence to thoroughly test cloud offering and ensure that these offerings can support a company's goals.

4.3.3.4 The Thin Line Between BI PaaS and SaaS

Based on our definition at the beginning of this paper, the primary difference between PaaS and SaaS is that PaaS normally represents a customizable cloud-based platform for application development, while SaaS provides an online application that has already been developed. In the case of cloud BI platforms (as well as some other cloud-based software), there is a blurry line between PaaS and SaaS classification. For example, Google's BigQuery BI offering can be considered both a SaaS and PaaS solution – it is SaaS for users who run queries against it, but PaaS for developers who use an API to program it.

4.3.3.5 When to Use BI in the Public Cloud

I recommend considering public data analytics and BI offerings in following scenarios:

- Data sources from which a company plans to analyze data are already hosted in the cloud. For example, if you are looking to integrate your BI platform with information from the Salesforce CRM system, then it makes sense to build a BI platform in the cloud because your data source is already in the cloud.
- An organization is willing to accept the limitations of a BI cloud provider and can clearly set expectations in terms of available features and potential cloud BI platform customizations needed.
- A company cannot predict computational workloads, and therefore will strongly benefit from cloud elasticity.
- A company doesn't have sufficient financial resources to build an internal BI platform, and therefore must rely on a public cloud provider that doesn't require up-front investment.
- The cloud provider's SLA (service-level agreement) fits a company's policies and needs.
- A company doesn't have the internal IT resources to deliver a solid BI platform.
- A company knows that all required data sources can be integrated with a cloud BI platform.

4.3.4 PaaS: Integration

Pricing: Depends on features – number of supported integration connections, security, user licenses, support options, data transformations, and others.

Cloud integration platforms help companies to integrate data from multiple sources. These sources may include on-premise private cloud databases, public cloud databases, third-party vendor databases, CRM systems, and messaging systems. Data integration is a very complex and tedious task, and data integration challenges include maintaining data quality, compatibility, and standards.

Dell Boomi is a leader in cloud data integration. Boomi started as an independent company, but was acquired by Dell in 2010. Boomi integrates over 40 different technologies and protocols, including those of Oracle, SAP, Siebel, Salesforce, Hadoop, and QuickBooks.

4.3.4.1 *Why use an integration service like Boomi instead of coding your own integration solutions?*

As shown in Figure 19, when using Boomi customers save a considerable amount of money on IT labor costs, maintenance costs, security, and upgrades.

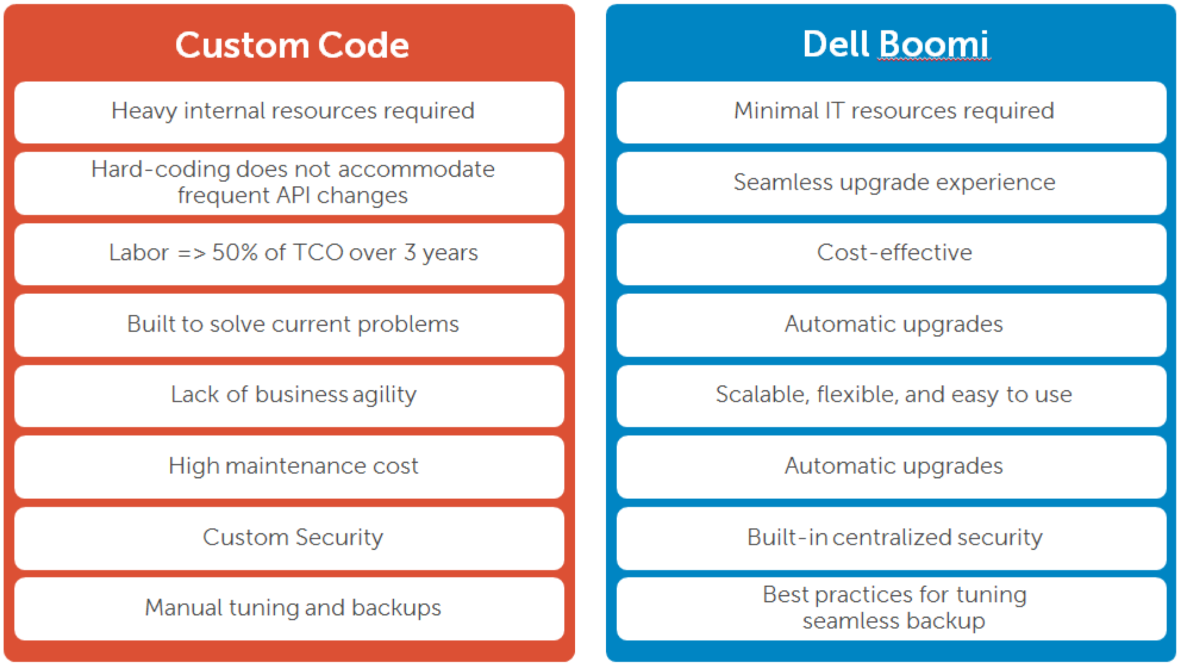


Figure 19: Boomi vs. Custom code [19]

4.3.4.2 *How does Boomi work?*

Boomi provides an easy-to-use integration workflow designer that has access to a library of integration connectors and maps. This application can be used to design data integration flow. Once integration workflow is designed and tested, it can be loaded to "Atom" – a runtime execution container similar to an application server. Atom can be deployed to the Dell Boomi cloud (see Figure 20), a private on-premise cloud, or a public third-party cloud. When Atom is deployed to a cloud, Boomi provides a set of monitoring and management tools for integration processes.

Integration runtime processes transfer data from a source to a destination, performing all specified mapping and formatting along the way. The structure and format of the source and destination data are represented by profiles, of which there are four basic categories: flat file, XML, EDI, and database. Data mapping and conversion operations typically include multiple data transformations between different categories of profiles.

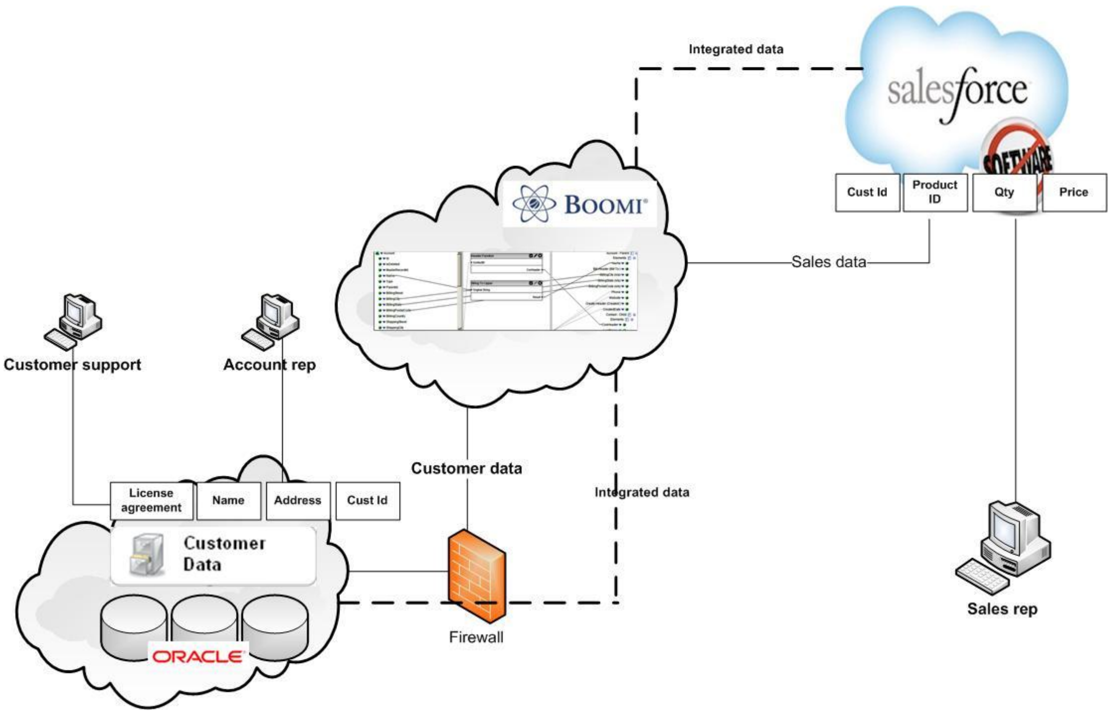


Figure 20: Boomi cloud integration example

In Figure 20 customer data stored in internal Oracle database is integrated with sales data stored in Salesforce CRM. Boomi PaaS integration software performs data mapping and aggregation.

4.3.5 PaaS: Development & QA

Pricing: varies by scaling, tools, supported runtimes, and SLA

PaaS Development and QA services help companies to improve code quality and development team collaboration, and to accelerate software development and a continuous integration cycle. Many development teams prefer to use the agile software development method [21] over the traditional waterfall model [22]. The agile development method assumes high level of collaboration and an intense continuous delivery cycle with frequent code releases, tests, and deployments. Modern development teams are often virtual (or otherwise geographically distributed) and therefore can benefit from PaaS tools that simplify and streamline team collaboration and create a faster time-to-market cycle.

As noted earlier, the software development process typically takes less time than subsequent software testing, maintenance, and support. This is a known problem, and a gap that many PaaS development services work to address. Most PaaS development services are oriented towards an agile development process; this is not to say that these PaaS services are not applicable to the waterfall method, but companies won't likely realize the same level benefits as with agile methodology.

Software development and operations include seven fundamental phases (see Figure 21): plan, code, build, test, release, deploy, and operate. In the agile development process, phases from code to release are highly iterative because code releases are very frequent and testing is continuous.

Many companies use agile methodology and deploy new releases of code several times a week, or even per day: "SlideShare deploys several times a day (between 2 and 20 times… It feels risky at first, but it's actually much less risky than a big bang deployment. If something goes wrong, it's easy to triangulate the cause, and if something doesn't go wrong, it's easy to trust that code" – Director, LinkedIn (SlideShare).

Many frequent small releases can be more efficient than one large cumulative release:

- Faster response to customer feedback about the product
- Small releases are easier to troubleshoot and roll back
- Sustainable development cycle
- Continuous motivation and attention from the team
- Quick adaptation to new requirements and trends

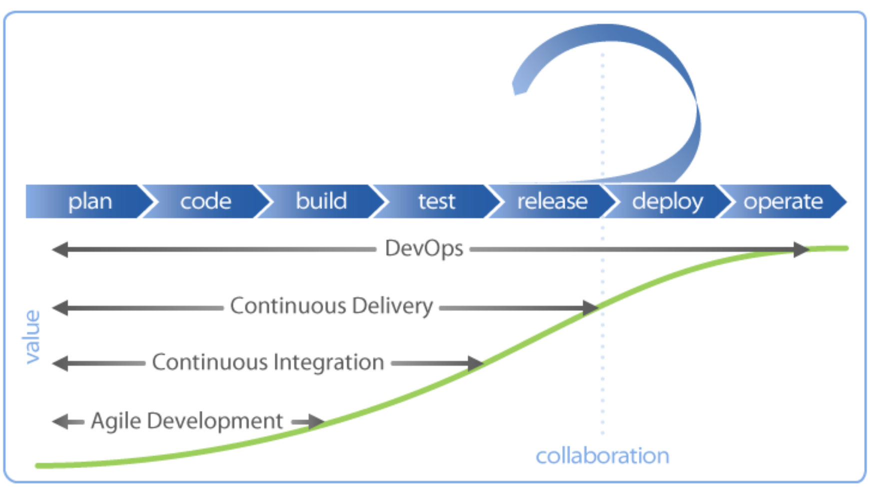


Figure 21: Software continuous delivery and operations processes [20]

In an enterprise organization, each of the development and operation processes has a separate employee role assigned – there are several employees with different functional roles normally involved in the software development and delivery process. However, such functional segregation of duties often delays the software delivery process, and raises the chances of inadvertent bugs and errors. In highly-segmented teams, development employees don't always feel fully responsible for their actions, and tend to leave specific functions to others. For example, developers may feel that they are not fully responsible for the quality of the code, and focus on pushing their code into the hands of QA engineers in the hope that all bugs will be caught by the subsequent team. Needless to say, such a strategy doesn't work well for the end product.

PaaS development platforms like Google AppEngine and Amazon Beanstalk take a continuous delivery approach to the next level by giving more power to developers through automating build, test, release, deployment, and operation processes. In Figure 22, we outline continuous delivery functional roles and compare these roles between enterprise on-premise and PaaS development environments.

Delivery phase	**Enterprise on-premise**	**Enterprise PaaS**	**Phase description**
Plan	Developer, QA, Operations	Developer	Software architecture and infrastructure design
Develop	Developer	Developer	Software development
Build	Build engineer	Automated by PaaS	Software packaging
Test	QA engineer	Partially automated by PaaS/ Developer	Software testing
Release	Build engineer	Automated by PaaS	Software release (version labeling, deployment preparation)
Deploy	Operations engineer	Automated by PaaS	Software package deployment to production infrastructure

Operate	Operations engineer	Automated by PaaS/Developer	Monitoring, Reporting, Troubleshooting, Failover

Figure 22: Software continuous delivery roles, on-premise vs. PaaS

Some may argue that not all software delivery and operations processes can be fully automated, and it is difficult to disagree with that. However, PaaS simplifies these processes to a level that software developers can handle. That is, you don't need to be an operating system expert or be closely familiar with network routing protocols in order to manage the operations of your infrastructure, because the cloud platform will do that for you. In addition to routine infrastructure operations, PaaS also auto-scales infrastructure based on certain application performance conditions. It takes substantial effort and expertise to design such automated scalability internally, while a PaaS platform provides it as part of the service offering.

Here, we review the two most popular development PaaS platforms – Google AppEngine and Amazon BeansTalk – and outline the differences between them.

4.3.5.1 Google App Engine

Google App Engine (GAE) is a pure PaaS platform that completely abstracts infrastructure services away from developers. Physically, GAE is represented by the web or application server, depending on whether the developer uses Python or Java. Google App Engine fully supports the Python, Java, and Go programming languages. GAE runs within a "sandbox," which isolates and secures operating system processes (see the nodes in Figure 23); so, the GAE web server doesn't have direct access to file system components like network sockets, system calls, and schedulers. Google wants to take on all infrastructure-related tasks, and therefore doesn't provide access to lower-level OS services. Instead, the service offers a rich set of higher-level APIs, (see Figure 23) to cover a majority of typical development needs. APIs provide programmable access to a wide set of services, such as SQL and NoSQL databases, Mail, MapReduce, and Log services.

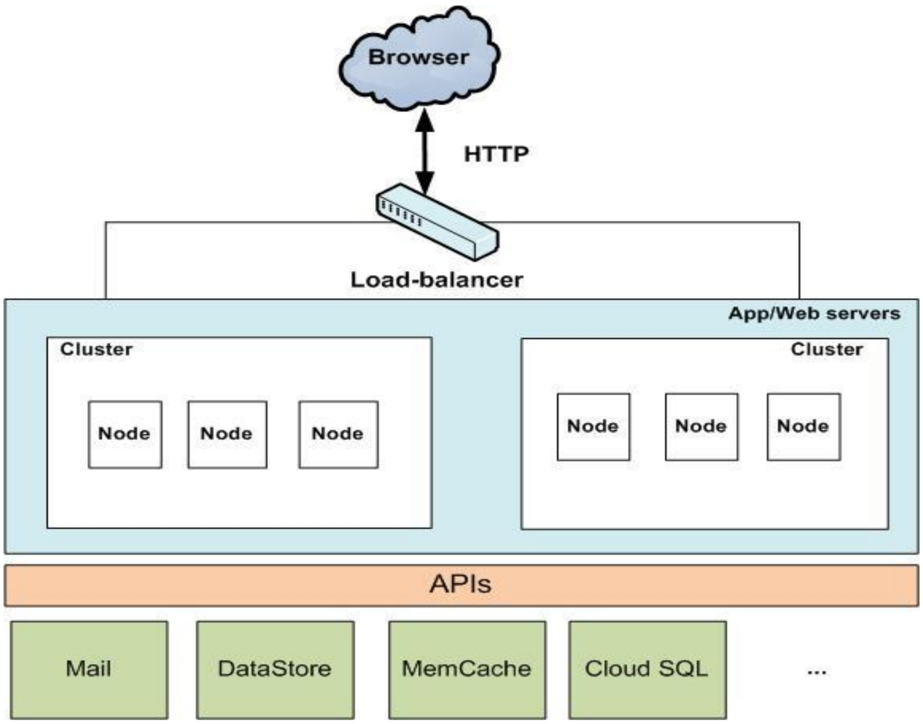


Figure 23: Google App Engine architecture

There are absolutely no system administration or other infrastructure tasks involved, as GAE handles all system routines. GAE continuously monitors application performance and auto-scales the environment by adding new nodes to the application cluster. All application-related configuration is done using configuration descriptor files, which are packages within the application and can be written in XML or YAML formats. A GAE application descriptor can configure a variety of features and tasks, such as security, job scheduling, task queuing, URL handling, database indexing, and backend server instances. Below is an example of a basic application configuration file:

```
application: myapp
version: 1
runtime: java

handlers:
- url: /admin/*
  login: admin
```

This YAML descriptor configures the “myapp” application, version 1, which runs in a Java container. It also defines a handler, who requires authentication to access/admin a URL, and defines that only an admin user can be authenticated.

4.3.5.2 Amazon Elastic Beanstalk

Amazon Elastic Beanstalk (AEB) is a PaaS platform built on top of Amazon’s IaaS infrastructure. AEB runs applications within EC2 cloud virtual machines (see Figure 24) – which we discussed in the IaaS: Computation section. The difference between Amazon IaaS offerings and PaaS AEB is that AEB automatically provisions the infrastructure and reduces the amount of time required for continuous integration tasks. At the same time, AEB provides full access to the operating system and other lower-level infrastructure components. AEB supports the Java, .NET, PHP, and Python programming languages.

To use AEB, you simply upload your application deployment file to the Amazon cloud using Amazon Web Services console or the Git version control repository, then specify which version of the application you want to deploy and which environment (OS, Application server, DB, etc.) you want to run this application against. Once the environment is launched, AEB automatically handles capacity provisioning, auto-scaling, health monitoring, and other infrastructure operations – customers needn’t spend time on infrastructure operations. However, if customers *want* to customize the infrastructure configuration, they have full access to all system components.

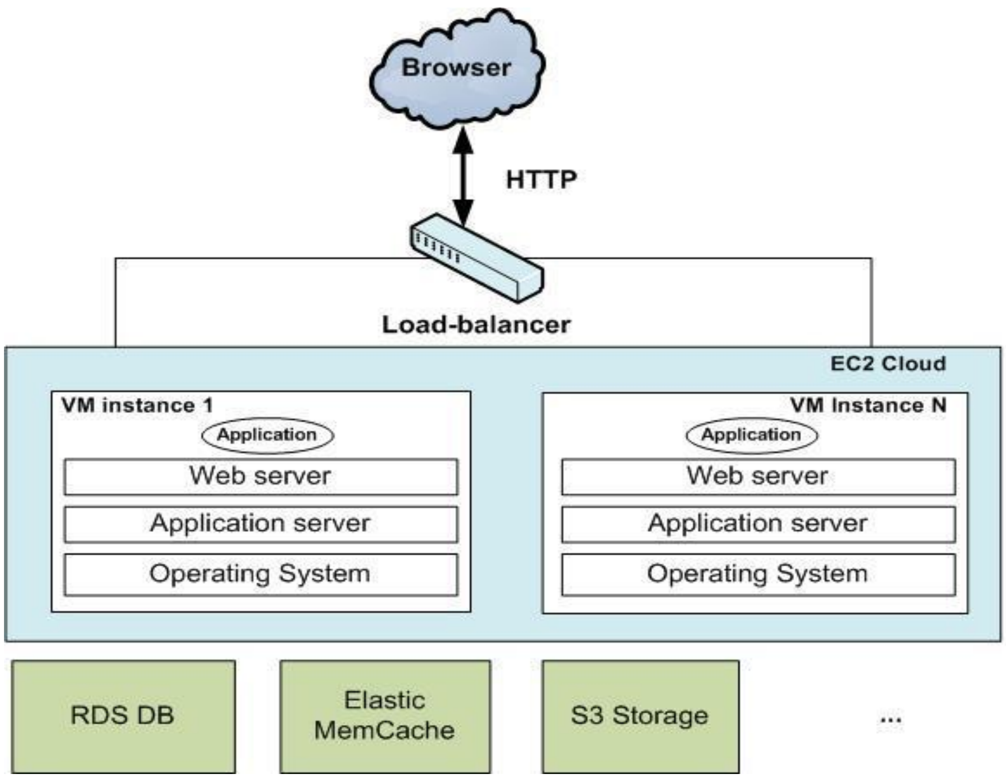


Figure 24: Amazon Elastic Beanstalk architecture

4.3.5.3 Amazon Beanstalk vs. Google App Engine

	Amazon Elastic Beanstalk	**Google App Engine**
Container type	Virtual Machine	Sandbox (process isolation)
Language support	Java, .NET, Python, PHP	Java, Python, GO, partial support for other languages
Customization	Fully customizable. Access to OS services, APIs, etc.	Limited customization. No access to OS services. No custom libraries, or APIs
Ease of use	Good. UI console, configuration descriptors, CLI. Easy to deploy.	Good. AWS UI console, CLI. Access to version control.
Features	All Amazon Web Services or any	Only services/APIs included with App Engine, but there are many of

	custom features are available	them and these are mostly sufficient
Pricing	EC2 pricing: $/number of hours VM running	$/number of apps, number of instances and features
Free	One EC2 micro instance for one year	Generous free quota per day: 28 instance hours, 1GB data storage, 1GB in/out bandwidth
Auto-scaling	Yes	Yes
Monitoring	Yes, UI tools and alerts	Yes, UI tools and alerts
Integration	Integration with any external and internal services	Integration with any Google services

Figure 25: Amazon Elastic Beanstalk vs. Google App Engine

Overall, both Amazon Elastic Beanstalk (AEB) and Google App Engine (GAE) provide solid PaaS offerings. These services have fundamentally different technical architectures: Amazon Beanstalk deploys applications within virtual machines, while Google App Engine runs applications within a sandboxed OS process. Both services are very easy to use, and customers can deploy applications within minutes.

From a **configuration** prospective, GAE might have a slight edge over AEB because it allows specifying all application and environment configuration details in application configuration descriptions. This approach is very common for developers, and requires no learning curve. The learning curve with AEB is potentially a little steeper because developers must understand at least the basics of Amazon Web Services in order to use AEB.

Beanstalk definitely wins in terms of **customization and available features**. Since it runs within a typical virtualization container, developers have full access to all services and applications as they would on any other server. GAE instead allows access to internal services only through the provided APIs. GAE prevents customers from accessing the OS directly, and from adding their own API libraries. This may create problems for customers who want to use custom language frameworks or do lower-level programming like thread management or network sockets. Although customization is definitely a limiting factor, Google nonetheless did a really good job of continuously expanding the list of available APIs and ensuring that these APIs cover most customer needs.

Integration is certainly an advantage of AEB. GAE can integrate only with Google services, but a customer seeking integration with a service that runs outside of Google will encounter challenges. If customers want to use GAE, they likely must move their entire web infrastructure to Google; however, as discussed previously, relying completely on a single vendor is considered a poor strategy because of lock-in potential.

GAE holds some advantage in **pricing** for small applications because of resource granularity – that is, customers can start very small and benefit from free GAE offerings. AEB is less flexible because it measures computational resources in virtual machines, and the smallest virtual machine costs approximately $40/month. For larger applications, GAE loses its price advantage.

4.3.5.4 When to use Google App Engine or Amazon Beanstalk?

I recommend using Google App Engine for small applications that don't have customization or integration requirements. Compared to Amazon Beanstalk, Google App Engine provides more granular pricing options because GAE charges customers per resource utilization, not per virtual machine as Amazon does. GAE's free quota is generous, and may even allow running small applications for free. Before moving application to Google App Engine, customers must be sure that GAE satisfies all application dependencies.

Amazon Elastic Beanstalk is a better fit for large enterprise applications that typically require integration with multiple services and advanced customizations.

4.4 SaaS services

Software as a service is a cloud services delivery model that offers an on-demand online software subscription. As with other cloud delivery models, SaaS offers companies the opportunity to reduce internal IT support costs and transfer maintenance liability to the SaaS provider. SaaS is by far the most widely-used cloud delivery model because almost every software vendor is looking to put its offering on the SaaS rails – there are SaaS offerings in every category of software products, and it would probably take days to list all SaaS software vendors in this paper. Therefore, in our SaaS taxonomy (see Figure 26) we list only selected groups of vendors in a few categories. In this section, we discuss SaaS trends using Salesforce as an example. Salesforce is the largest SaaS company, with $2.3 billion in revenue for 2012, representing approximately 15% of total SaaS market revenues ([24] "Gartner Says Worldwide Software-as-a-Service Revenue to Reach $14.5 Billion in 2012").

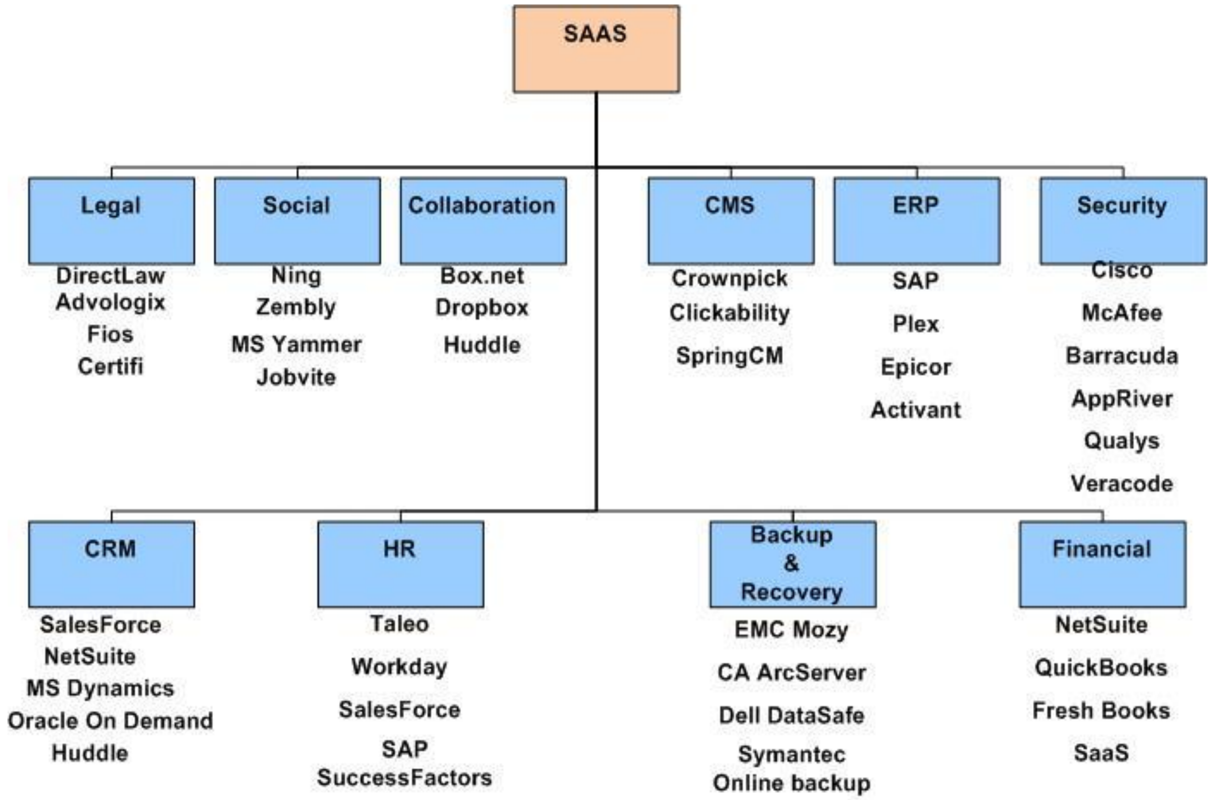


Figure 26: SaaS services taxonomy, selected categories and vendors

SaaS is a large market with strong growth potential. Gartner Group estimates that by 2015 worldwide SaaS revenue will reach $22.5 billion [24]. A continuation of the ASP (application services provider) model, the mainstream SaaS services appeared at the end of the 1990s when companies like Salesforce offered customers outsourced hosting and management of their software. The ASP model also offered centralized third-party application hosting, but was different from SaaS because hosting and operations still required manual involvement. Further, ASPs sometimes required the installation of a thick desktop client [44](locally running software which performs majority of the computational tasks on the user's computer, not on the server side), while SaaS applications are normally accessed through web browsers or mobile apps.

SaaS platforms employ multi-tenant architectures where the same platform hardware and software is shared among multiple customers. Multi-tenancy hosting and service automation allows SaaS providers to keep service costs low. Depending on the SaaS provider and type of application, some application customization is often available, but this is typically more limited than with PaaS and IaaS cloud solutions.

Initial SaaS ideas caused significant concern related to security, performance, and service availability. However, through the years SaaS technology and business models have significantly matured, and overcome many of these initial worries. Companies now actively use SaaS, and that's unlikely to change.

4.4.1 SaaS Business Challenge

The main competitive advantage fueling SaaS growth is the ability of SaaS providers to offer high-quality services at cheaper prices than on-premise software vendors. In addition, businesses are attracted to SaaS services because such expenses are treated as operational (OpEx) instead of capital (CapEx). As discussed in previous chapters, companies prefer OpEx over CapEx for the tax advantage.

The SaaS business model may seem perfect – it leaves all control over application management in the hands of SaaS provider. As such, the provider decides when to upgrade software, when to add new features, and when to raise prices.

There is no free lunch, however, and the cost advantage of SaaS business models comes at a price. Lock-in potential can be high with some popular SaaS services, such as CRM or ERP – those familiar with CRM or ERP systems know how difficult it is to switch vendors. Further, many SaaS companies – even those with thousands of customers – may not reach profitability for years after starting the service. The largest SaaS company, Salesforce, had negative net profit margins in the recent quarter (Q2, 2012). As a matter of fact, Salesforce hasn't had a quarterly profit margin higher that 10% (see Figure 27) since 2006, even though company revenues have been steadily growing. By comparison, SAP profit margin in Q2, 2012 was around 17%.

Figure 27: SAP vs. Salesforce, revenues and net profit margin (%)

So, why would an SaaS giant like Salesforce with 2012 revenues of $2.3 billion per quarter have a negative profit margin? The reason is **high customer acquisition costs** (i.e., sales and marketing expenses).

The SaaS business model follows the economy-of-scale principle – it is a low-price, high-volume business. Salesforce currently has roughly 90,000 business customers, a huge number. To maintain sustainable growth for their business, Salesforce must keep customer churn rates low and continuously accelerate the customer acquisition pace. They must also keep their prices low in order to compete with on-premise CRM vendors, other SaaS players, and new market entrants.

Let's compare SAP and Salesforce's financial income statements from fiscal year 2012 (***Source: http://investing.businessweek.com***).

	SAP (SAP)	Salesforce (CRM)
Revenues	14.3 billion (EUR)	2.26 billion (USD)
Cost of Goods (% of revenue)	28%	21%
Sales, marketing, and administration	**26%**	**66%**
R&D expenses	13%	13%
EBT(earnings before tax)	31%	- 1.5%
NET income	**24%**	**-0.5%**

Figure 27: SAP vs. Salesforce, income statement

In Figure 27, we can clearly see that "sales and marketing" represents a majority of Salesforce's expenses. R&D expenses are approximately equal (in terms of % from revenue) between two companies. Cost of goods is lower for Salesforce, which makes sense because their web-based solution is cheaper to develop and support compared to on-premise installed software. Salesforce also operates in a centralized, web-based, multi-tenant infrastructure that doesn't carry some of the expenses typical of on-premise software such as multi-platform support or integration with a variety of internal infrastructure systems.

Could Salesforce have survived and become so successful without such a massive investment in sales and marketing?

SaaS and the cloud in general is a relatively new market – it takes time to educate customers and establish a level of trust. Competition is fierce, and although barriers to enter the SaaS market are higher than those of IaaS, they are still not high enough to keep competitors out; SaaS providers must keep prices low to stay competitive. They can minimize their product development and delivery expenses, but at the same time they don't have the same profit opportunities in consulting and professional services as on-premise software vendors do. Large software vendors typically generate 30-40% of their revenues from services, while Salesforce revenues from services make only about 7-8%.

Salesforce will certainly have to reduce their investment in sales and marketing at some point when the SaaS market matures and consolidates.

4.4.2 Salesforce Platform Overview

Salesforce.com started as a CRM application in 1999, but with time it grew into a powerful enterprise platform covering many enterprise business and social needs. Salesforce components are closely integrated with each other, and all of them offer API for automation and external integration. The Salesforce CRM system has an easy-to-use web interface for customers, and an API for developers. Salseforce Force.com in particular is a PaaS development platform that natively supports several programming languages, including Java, PHP, .NET, and Ruby. It can be integrated with most popular cloud services, such as Amazon Web Services, Facebook, and Twitter.

The Salesforce platform (see Figure 28) consists of several components:

Sales cloud: A CRM (customer relationships management) system, which helps to organize customer information and manage sales processes. It helps to close deals faster, to research sales data, and to share all information with members of your team.

Service Cloud – customer service software that allows communication with customers through the contact center and social media

Radian6 – a social media marketing platform for customer engagement and marketing-lead generation

AppExchange – Salesforce's application store. It hosts applications developed by the Salesforce community.

Chatter – the company's private social network, which allows for quickly collaborating with colleagues, discovering resources, and sharing information and content.

Force.com – a PaaS development platform for creating and deploying Salesforce-integrated applications to the Salesforce cloud

Heroku – a cloud PaaS hosting platform that can be used to host any application

Data.com – a business data provider and aggregator

Database.com – PaaS database technology with a set of included APIs and security features

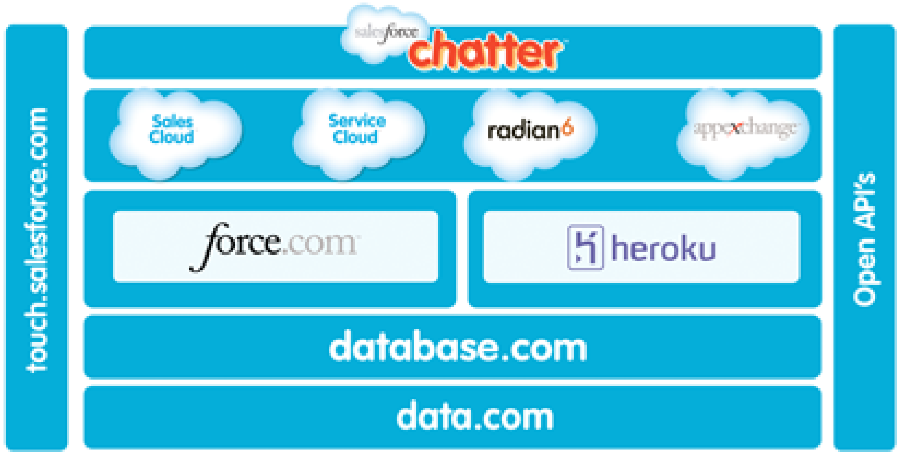


Figure 28: Salesforce's platform

4.4.3 Is Salesforce both SaaS and PaaS?

Salesforce has both SaaS and PaaS components; however, as with many other technologies, the line between SaaS and PaaS is very thin. It depends not only on technical characteristics, but also on user perception. From the end-user's perspective, Salesforce is certainly a SaaS because it has a nice user interface accessible through a browser or a mobile application. From development prospective, though, Salesforce is a PaaS because it provides programmable access through its API.

5 CONCLUSIONS

5.1 The Future of the Cloud Computing Market

Cloud Computing is a disruptive methodology that is rapidly changing how computing is done. When mass adoption of cloud computing services began in 2005 and 2006, several cloud providers achieved an early market lead. Then the market started to boom around these cloud services, and many companies saw potential and entered the market. Currently, although a few major players lead the market, none of them holds a market lock in terms of technology standards and features. According to James M. Utterback's *Mastering the Dynamics of Innovation* [14], this is a typical market response to the introduction of a new innovative product.

So, what should we expect to happen next in the cloud computing market?

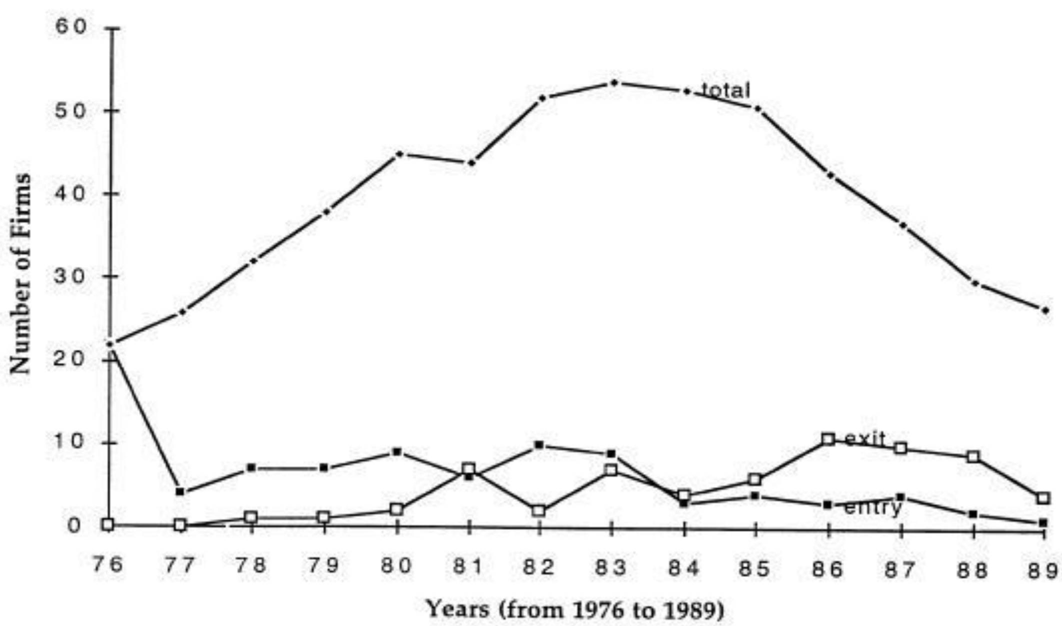
Utterback presents evidence for his hypothesis that the peak in the total number of competing market players occurs shortly after the "dominant design" state of a product is reached. Following the emergence of the dominant design, we can expect a significant number of market exits and consolidations.

By Utterback's definition, the dominant design of a product is "the one that wins the allegiance of the market place, the one that competitors and innovators must adhere to if they command significant market following" [14, p.24]. Once a dominant design is determined by market participants, companies tend to use it as a reference standard for their product implementations. Although there are can be some edge cases, and the dominant design may not necessarily satisfy the entire population of customers, it nonetheless typically covers the needs of the majority. The establishment of a dominant design cannot always be predicted – it depends on a combination of certain technical and business market events in a given timeframe.

To analyze current the cloud computing technology market and hypothesize about future trends, we can draw a historical parallel to compare the disruption caused by cloud computing technology with the past market entrances of other disruptive products.

Introduced by IBM 1973,**Winchester disk drives** were a disruptive technology for the rest of the decade. As Clayton Christensen presented in *The Innovator's Challenge: Understanding the Influence of Market Environment on Processes of Technology Development in the Rigid Disk Drive Industry* [15], there were a total of five generations of Winchester drives: 14-inch, 8-inch, 5¼-inch, 3 ½-inch, and 2½-inch. In the period from 1976 to 1983, many companies followed IBM's lead and entered the lucrative Winchester drive market.

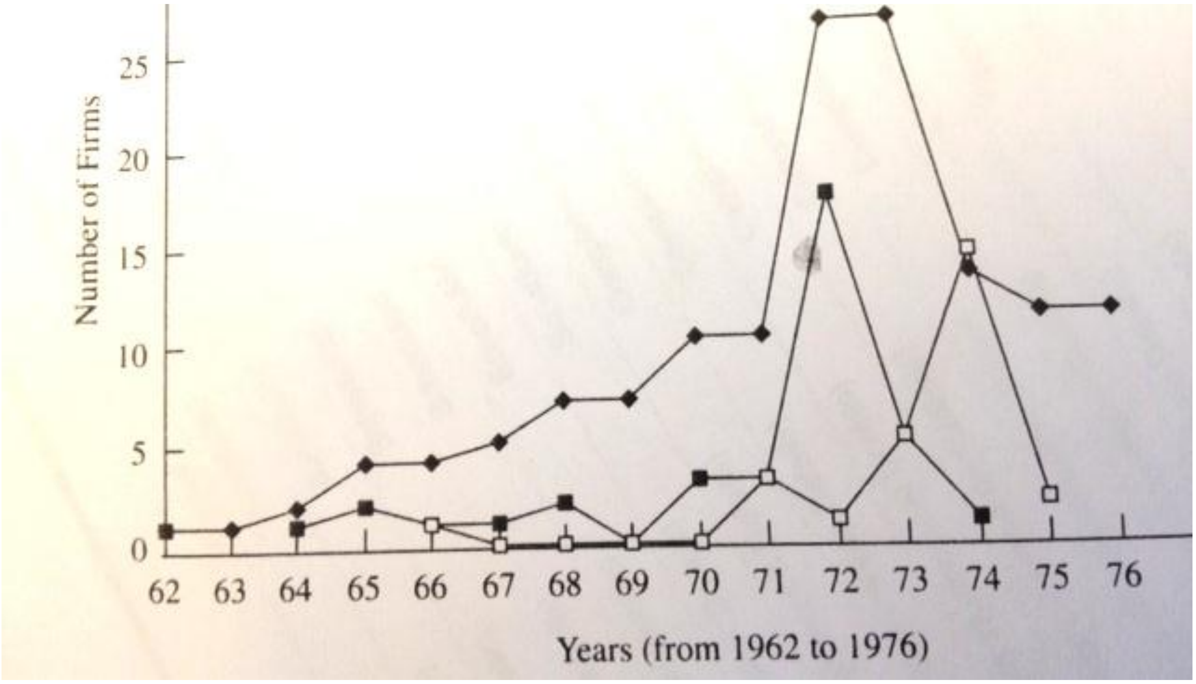
In the first few years, the number of market exits was minimal. The number of companies peaked around 1983, approximately when 3½-inch hard drives began to dominate the market. Once this dominant technology was determined, market exit events accelerated, followed by a number of consolidations and bankruptcies. By the 1990s, the market had standardized on 3½-inch drives, with only a few major vendors surviving the competition.



□ **EXIT** ■ **ENTRY** ● **TOTAL**

Figure 18: Number of firms in U.S. Winchester Disk Drive Industry [15]

The **electronic calculator** demonstrated similar market trends [14]**.** The market peaked in 1971-1972 with 25 participants. After that, exits started because a dominant design was introduced with the invention of the innovative calculator chip, which tremendously simplified the calculator-assembly process and minimized the number of parts.



□ **EXIT** ■ **ENTRY** ◆ **TOTAL**

Figure 19: Number of firms in the U.S. electronic calculator industry [16]

Both the Winchester drive and electronic calculator demonstrated the same pattern: the initial introduction of a disruptive product attracted multiple players, but after a dominant product design was chosen by market participants companies that couldn't adopt this design either left the market or merged with other players. It is worth noting that the dominant product design doesn't always represent the most sophisticated cutting-edge solution – it is simply a solution that fits the majority of customers and provides reasonable investment returns to vendors.

5.1.1 Public Cloud Domination

Based on market research [6] [11], we can hypothesize with a high level of confidence that the **public cloud computing** model is likely to become dominant for serving computing resources. According to the market survey [6], in 2011 roughly 28% of surveyed companies employed public cloud services, and by 2014 that share of companies using a public cloud will grow to 51%. That said, this doesn't mean that these companies will stop using private clouds; but, in the long term, private and hybrid cloud market shares are likely to continuously drop, and these cloud models will likely be used only for specific business cases.

At this point, cloud computing competition is heating up – we can assume that the market hasn't yet peaked, but given the high number of market players it is obviously getting closer to that point. Following the market peak, we should expect a large number of consolidations and market exits. After that, the cloud computing market structure should stabilize, with only a few remaining large companies dominating the market.

The current state of cloud computing technology is certainly not final. Cloud technology and related business models will evolve further, and undergo several iterations of product innovation followed by an even longer period of process innovation. Cloud computing standards are still under development, and it will take some time to develop universal standards.

5.1.2 Open-Source vs. Proprietary Cloud Technologies

Many new cloud market entrants prefer to use open-source technology to enable their platforms. However, more established players like Amazon, Google, and Microsoft employ proprietary systems. The large companies that took an early cloud market lead developed sophisticated proprietary cloud technologies to gain competitive advantage. It is not feasible for these companies to open-source their technology and give away their technical lead. New market entrants are in the different position of not having time to develop their own technologies to catch up with the market. One of the major companies using open-source cloud platforms is Hewlett-Packard, which just recently entered the public cloud market in 2012. To quickly build their cloud platform and catch up with other market players, they use the OpenStack open-source cloud platform.

In the long run, we expect to see both proprietary and open-source technologies competing in the cloud market. Potentially, some of the proprietary technologies could be open-sourced later on, once the market has consolidated and stabilized. At the end-market stabilization period, companies will pay more attention to process innovation rather than product innovation. (**Product innovation** is the creation of new products or services, or significant improvement of an existing product or service. **Process innovation** means minor changes or improvements in product or service capabilities, such as a new service delivery method, software feature, and so on.)

5.2 Cloud Delivery Models

For the foreseeable future, all three major delivery models (IaaS, PaaS, and SaaS) are here to stay. It is likely that IaaS market share is going to steadily drop as customers realize that there is more value and resource-savings from software and platform services rather than infrastructure. However, certain technical factors and dependencies will keep IaaS cloud services afloat for some time to come.

We will likely see more consolidations in the IaaS space because the IaaS cloud service market has a very low barrier to entry and minimal technological differentiation factor. As IaaS becomes a commodity, investment returns will drop and cloud providers will be forced to move to the more lucrative PaaS and SaaS segments. Smaller IaaS providers do not have the financial resources to develop sophisticated software to compete in the PaaS or SaaS space, and therefore consolidations and partnerships will be necessary for survival.

PaaS development and QA services help companies to improve code quality, foster development team collaboration, and accelerate software development and a continuous integration cycle. Many development teams prefer to use the agile software development method [21] over the traditional waterfall model [22]. PaaS development platforms and collaboration tools provide companies with substantial time and money savings, and therefore we are likely to see more innovation in this segment.

Cloud business intelligence and data analytics remain one of the most untapped segments in the cloud services market. The primary cloud BI adoption challenge is security.

Several key factors are driving BI into the cloud:

- Higher time-to-value ratio
- OpEx tax structure
- Lack of skills and internal expertise
- Elasticity and pay-as-you-grow options
- Lower maintenance costs

I recommend considering public data analytics and BI offerings in following scenarios:

- Data sources from which the company plans to analyze data are already hosted in the cloud. For example, if you are looking to integrate your BI platform with information from Salesforce's CRM system, then it makes sense to build a BI platform in the cloud because the data source is already there.

- The organization is willing to accept the limitations of a chosen BI cloud provider and can clearly set expectations in terms of needed features and potential cloud BI platform customizations.
- The company cannot predict its computational workloads, and therefore will strongly benefit from the cloud's elasticity.
- The company lacks the financial resources to build an internal BI platform, and therefore must rely on a public cloud provider that doesn't require an up-front investment.
- The cloud provider's SLA (service-level agreement) fits the company's policies and needs.
- The company lacks the internal IT resources to deliver a solid BI platform.
- The company knows that all required data sources can be integrated with a cloud BI platform.

5.2.1 The Thin Line between IaaS, PaaS, and SaaS

By definition, cloud delivery models should be classified by the following characteristics.

SaaS: software applications (not including ones with APIs)

PaaS: execution runtimes without direct access to the OS (databases, development platforms, and application servers)

IaaS: virtual machines, servers, storage, network gear, and any other hardware/OS

In some cases it is difficult to precisely classify a specific cloud computing service. For example, Amazon RDS database services can arguably be attributed to either IaaS or PaaS delivery models. As providers looking to offer better productivity tools for their customers, they are complementing IaaS with PaaS and SaaS services. Therefore, the line between cloud delivery models is becoming blurred.

Salesforce offers both SaaS and PaaS components. From an end-user's perspective Salesforce is certainly a SaaS because it has nice user interface accessible through a browser or mobile application. However, from a development prospective Salesforce is a PaaS because it provides programmable access through an API. Similarly, in the case of cloud BI platforms Google's BigQuery BI offering could be considered both a SaaS and PaaS solution – it is SaaS for the users who run queries against it, but PaaS for developers who use an API to program it.

As such, we can conclude that cloud delivery model classifications depend not only on technical characteristics, but also on user perception.

5.2.2 Cloud Adoption and Control Challenges

Cloud transformation is a lengthy process that involves both technical and organizational challenges. As with the introduction of other disruptive technologies, cloud computing has experienced considerable adoption resistance, and cloud proponents should be prepared to overcome resistance in their organizations. Complexity paired with uncertainty creates organizational cloud adoption barriers.

Security is by far the most serious barrier to the adoption of cloud computing, but there are other adoption challenges as well:

- Data security concerns
- Cost uncertainty
- Loss of control
- Reliability concerns
- Reduced data portability
- Reduced software compatibility
- Reduced performance
- Lock-in fears

Organizational challenges are hard to quantify, but such challenges can be critical in the decision-making process.

5.3 Cloud Services Pricing

Cloud market competition is becoming intense, and providers are pressed to aggressively reduce service prices. A competitive and uncertain market with many existing market players and continuous market entrances leads to high market volatility. In such a volatile market, long-term customer commitments are key for a company's success. Cloud providers offer lucrative deals to lock customers in because of these benefits:

- With long-term customers commitments, providers can better plan their cloud capacity for years ahead. The ability to forecast capacity helps providers to make correct capital expenditure (CapEx) decisions.
- There are low barriers to enter some cloud market segments (such as IaaS), and entrants periodically disrupt the market by offering new technologies and competitive processes. This creates a challenge for established cloud providers, and makes it more difficult to keep customers. Prepaid services ensure long-term customer commitment.
- Providers want to prevent customers from using the services of other providers.

Therefore, in the current market customers can secure much better prices if they commit to a provider (see Appendix A). In many cases, the price difference between pay-as-you-go and prepaid pricing can be very large.

The SaaS model follows the economy-of-scale principle – it is a low-price, high-volume business. The customer acquisition process is expensive. Salesforce and other SaaS providers have low profit margins because they must invest heavily in the sales and marketing of their products. They will have to reduce this investment in sales and marketing at some point when the SaaS market matures and consolidates.

To provide sustainable growth for their business, SaaS providers must keep customer churn rates low and continuously accelerate the customer acquisition pace. They must also keep their prices low to compete with on-premise CRM vendors, other SaaS players, and new market entrants.

6 REFERENCES

[1] PARKHILL, D. The Challenge of the Computer Utility. Addison-Wesley Educational Publishers Inc., US, 1966.

[2] HAMILTON, J. Internet-Scale Service Efficiency. In Large-Scale Distributed Systems and Middleware (LADIS) Workshop (September 2008).

[3] Above the Clouds: A Berkeley View of Cloud Computing. Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia UC Berkeley Reliable Adaptive Distributed Systems Laboratory http://radlab.cs.berkeley.edu/ February 10, 2009

[4] Architecting for the Cloud: Best Practices, *Jinesh Varia , January 2010*

[5] http://en.wikipedia.org/wiki/Cloud_computing

[6] "Cloud Computing Takes Off", Morgan Stanley, May 23, 2011

http://www.morganstanley.com/views/perspectives/cloud_computing.pdf

[7] http://aws.amazon.com/solutions/case-studies/acquia/

[8] http://www.colt.net/cio-research/index.html

[9] "Eight fundamental truths about cloud", Bob Deutsche, Intel corp., 2011

[10] http://aws.amazon.com/ec2-sla/

[11] http://gartner.com

[12] http://scalr.com

[13] http://www.rightscale.com/products/mycloud.php

[14] Utterback, James M., "Mastering the Dynamics of Innovation", Harvard Press, 1994

[15] Christensen, Clayton M., *The Innovator's Challenge: Understanding the Influence of Market Environment on Processes of Technology Development in the Rigid Disk Drive Industry*. Unpublished doctoral dissertation, Harvard University Graduate School of Business Administration, Boston, Massachusetts, 1992

[16] B. Majumdar, Innovations, Product developments and Technology Transfer: An Empirical Study of Dynamic Competitive Advantage, The Case of Electronic Calculator, Ph. D, Diss. Case Western Reserve University, Cleveland, Ohio, 1977

[17] http://en.community.dell.com/techcenter/b/techcenter/archive/2012/02/29/digging-deep-into-dell-boomi-how-does-it-work.aspx

[18] http://www.gartner.com/it/page.jsp?id=1903814

[19] http://en.community.dell.com/techcenter/b/techcenter/archive/2012/02/29/digging-deep-into-dell-boomi-how-does-it-work.aspx

[20] collab.net

[23] http://www.slideshare.net/rajdeep/introduction-to-google-app-engine-presentation

[24] http://www.gartner.com/it/page.jsp?id=1963815

[25] http://salesforce.com

[26] http://www.zdnet.com/blog/open-source/amazon-ec2-cloud-is-made-up-of-almost-half-a-million-linux-servers/10620

[27] http://www.quora.com/Amazon/How-and-why-did-Amazon-get-into-the-cloud-computing-business

6.1 Definitions

[21] http://en.wikipedia.org/wiki/Agile_software_development
[22] http://en.wikipedia.org/wiki/Waterfall_model
[31] http://en.wikipedia.org/wiki/Denormalization
[32] http://en.wikipedia.org/wiki/Service-level_agreement
[33] http://en.wikipedia.org/wiki/Operational-level_agreement
[33] http://en.wikipedia.org/wiki/IP_address
[34] http://en.wikipedia.org/wiki/Access_control_list
[35] http://www.ehow.com/facts_7523483_ssl-termination.html
[36] http://en.wikipedia.org/wiki/Load_balancing_(computing)
[37] http://en.wikipedia.org/wiki/Virtual_IP_address
[38] http://en.wikipedia.org/wiki/IPv6
[39] http://en.wikipedia.org/wiki/Bandwidth_throttling
[40] http://aws.amazon.com/autoscaling/
[41] http://en.wikipedia.org/wiki/Job_scheduler
[42] http://en.wikipedia.org/wiki/DNS
[43] http://en.wikipedia.org/wiki/Backup
[44] http://en.wikipedia.org/wiki/Thick_client
[45] http://en.wikipedia.org/wiki/Data_analysis
[46] http://aws.amazon.com/autoscaling/