Clarification of Cloud Computing

Zhongliu Xie
University of Nottingham
Jubilee Campus, Wollaton Road
Nottingham, UK
+447879757914
cloud.arkadia@gmail.com

ABSTRACT

Cloud computing is a controversial term. Various advertisements, articles and speeches have explained cloud computing with dramatically distinct definitions. Today, the term "cloud computing" has been too much overused and misused, causing confusions and controversies to a significant extent, and jeopardizing the progress of relevant research. Cloud computing involves not only computing technologies, but also business issues. This paper aims to clarify the concept of cloud computing, from both technological and business perspectives, addressing the controversies, in order to help with the progress of relevant research.

Keywords

Cloud computing, parallel processing, concurrency, distributed system, client-server computing

1. INTRODUCTION

Cloud computing is a new concept emerged in the past decade, and has kept increasingly drawing social attentions throughout the world. It is a very complicated and controversial concept. Different researchers, groups and organizations hold much different point of views towards its definition and relevant aspects. Various advertisements, articles and speeches have explained cloud computing with dramatically distinct concepts. Some researchers treat it as an evolutional technology, some researchers perceive it as an innovative business approach, some equate it with the increasingly popular concept "Software as a Service", and some others treat it as nonsense, as a business shunt or even conspiracy.

In fact, cloud computing involves both computational technologies and business issues. At the first glance, the idea of cloud computing could be superficially introduced by comparing it with the traditional computing: With traditional computing, software programs are installed and run on individual computers, where relevant operations, which could be generalized into a collection of data processing and storing activities, are carried by CPU, Memory, Hard disk and so on. Nevertheless, with cloud computing, the programs are not run on the local computers, but on separate ones from integrated clusters of computers to which one access through the Internet. These computer clusters are resembled as 'cloud' by Ramnath, who was the first researcher providing a formal academic definition of cloud computing in 1997.

Many people believe that the development of computing would probably resemble that of electricity: Right after Faraday's discovery of electro-magnet, the supply of electricity was provided by individual power generators; but when Tesla invented the AC power model and the remote transmission methods, electricity became a type of utility under massive production to benefit from Economies of Scale. Later on the significant contribution of electricity to human life made it become the key element of the Second Industrial Revolution. In the IT Revolution, PCs are similar to the individual power generators in households, and computing cloud resembles those integrated power plant clusters. It is believed that computing will finally become a type of utility like electricity and gas, where households only use the terminal devices with network access to the cloud nexus, to benefit from inexhaustible supply of utility computing. This is why cloud computing is believed to play a key role in the next generation of IT Revolution

Section 2 will describe four major technological features of cloud computing, which are respectively parallel processing, concurrency, distributed system and client-server model. Section 3 will discuss the business issues involved in cloud computing, with reference to theories of economics, and analyze the three prevailing business operations (SaaS, PaaS & IaaS) in more detail. Section 4 will assess the different perspectives of cloud computing from different researchers, and particularly make a comparison between two groups: enterprise practitioners and scholars. Finally Section 5 will give the conclusion and Section 6 some future work.

2. TECHNOLOGICAL PERSPECTIVE

2.1 Parallel Processing

Assuming there are five jobs to be done. In a uni-processor non-preemptive environment, these jobs need to be handled one by one, which is often called as sequential processing. In a multi-processor environment, different jobs can be handled simultaneous by the number of processors available, achieving:

$$\text{Efficiency} = \text{individual efficiency} \times \text{No. of processors}$$

This is referred to as parallel processing. Parallel processing can be realized at five levels: Sub-program level, Statement level, Operation level and micro-operation level, as shown [19]:

ICHIT'10, August 26–28, 2010, Daejeon, Korea.

Clarification of Cloud Computing

Zhongliu Xie
University of Nottingham
Jubilee Campus, Wollaton Road
Nottingham, UK
+447879757914
cloud.arkadia@gmail.com

ABSTRACT
Cloud computing is a controversial term. Various advertisements, articles and speeches have explained cloud computing with dramatically distinct definitions. Today, the term "cloud computing" has been too much overused and misused, causing confusions and controversies to a significant extent, and jeopardizing the progress of relevant research. Cloud computing involves not only computing technologies, but also business issues. This paper aims to clarify the concept of cloud computing, from both technological and business perspectives, addressing the controversies, in order to help with the progress of relevant research.

Keywords
Cloud computing, parallel processing, concurrency, distributed system, client-server computing

1. INTRODUCTION
Cloud computing is a new concept emerged in the past decade, and has kept increasingly drawing social attentions throughout the world. It is a very complicated and controversial concept. Different researchers, groups and organizations hold much different point of views towards its definition and relevant aspects. Various advertisements, articles and speeches have explained cloud computing with dramatically distinct concepts. Some researchers treat it as an evolutional technology, some researchers perceive it as an innovative business approach, some equate it with the increasingly popular concept "Software as a Service", and some others treat it as nonsense, as a business shunt or even conspiracy.

In fact, cloud computing involves both computational technologies and business issues. At the first glance, the idea of cloud computing could be superficially introduced by comparing it with the traditional computing: With traditional computing, software programs are installed and run on individual computers, where relevant operations, which could be generalized into a collection of data processing and storing activities, are carried by CPU, Memory, Hard disk and so on. Nevertheless, with cloud computing, the programs are not run on the local computers, but on separate ones from integrated clusters of computers to which one access through the Internet. These computer clusters are resembled as 'cloud' by Ramnath, who was the first researcher providing a formal academic definition of cloud computing in 1997.

Many people believe that the development of computing would probably resemble that of electricity: Right after Faraday's discovery of electro-magnet, the supply of electricity was provided by individual power generators; but when Tesla invented the AC power model and the remote transmission methods, electricity became a type of utility under massive production to benefit from Economies of Scale. Later on the significant contribution of electricity to human life made it become the key element of the Second Industrial Revolution. In the IT Revolution, PCs are similar to the individual power generators in households, and computing cloud resembles those integrated power plant clusters. It is believed that computing will finally become a type of utility like electricity and gas, where households only use the terminal devices with network access to the cloud nexus, to benefit from inexhaustible supply of utility computing. This is why cloud computing is believed to play a key role in the next generation of IT Revolution

Section 2 will describe four major technological features of cloud computing, which are respectively parallel processing, concurrency, distributed system and client-server model. Section 3 will discuss the business issues involved in cloud computing, with reference to theories of economics, and analyze the three prevailing business operations (SaaS, PaaS & IaaS) in more detail. Section 4 will assess the different perspectives of cloud computing from different researchers, and particularly make a comparison between two groups: enterprise practitioners and scholars. Finally Section 5 will give the conclusion and Section 6 some future work.

2. TECHNOLOGICAL PERSPECTIVE

2.1 Parallel Processing
Assuming there are five jobs to be done. In a uni-processor non-preemptive environment, these jobs need to be handled one by one, which is often called as sequential processing. In a multi-processor environment, different jobs can be handled simultaneous by the number of processors available, achieving:

$$\text{Efficiency} = \text{individual efficiency} \times \text{No. of processors}$$

This is referred to as parallel processing. Parallel processing can be realized at five levels: Sub-program level, Statement level, Operation level and micro-operation level, as shown [19]:

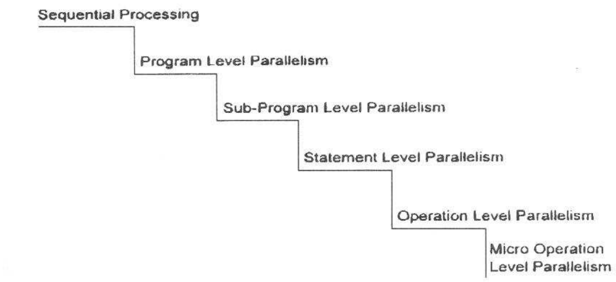

Figure 1. Levels of paralel processing

There are various approaches of classifying architectures that realize parallel processing, among which, Flynn's Taxonomy is the most widely-used one. This taxonomy is based on data and instruction, and contains four architectures, i.e. Single Instruction Single Data (SISD), Multiple Instruction Single Data (MISD), Single Instruction Multiple Data (SIMD) and Multiple Instruction Multiple Data (MIMD) architecture, as shown:

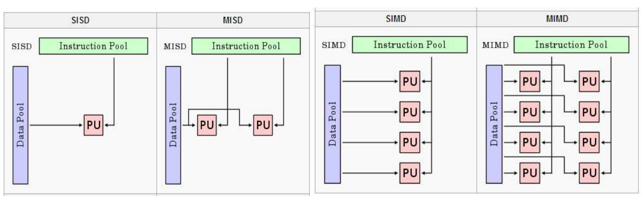

Figure 2. Flynn's Taxonomy

(Source: http://en.wikipedia.org/wiki/Flynn's_taxonomy)

Researchers have pointed out the incompleteness of this taxonomy by providing exceptions such as pipelining architecture, which realize parallelism but cannot be classified in any of the above four groups.

Parallel processing would encounter the problems of data dependency and decomposability. Decomposability refers to atomic function that cannot be divided into sub-function; while data dependency implies an intrinsic sequential characteristic, in other words, there are several steps of calculation need to be performed in a specific order [19]. Furthermore, based on the communication mechanisms between the processors, SIMD and MIMD architectures could be further divided into two categories: using shared memory or through network. The shared memory approach is about multiple processors sharing a common memory, and is usually the case of using multicore processors.

2.2 Concurrency

Today, every operating system is able to run various programs such as browsing websites, displaying PDF files and playing some audio files at the same time. This is called multi-task processing. Multi-thread processing employs this idea at sub-program level, and allows multiple related processes to work together to carry out specific tasks. Concurrency is the combination of both multi-tasking and multi-threading. The most fundamental property is ensuring multiple programs and sub-programs running in harmony when they are overlapped in time, whether or not they are communicating. In that sense, the calculations in two processes do not necessarily need to be performed in parallel. In other words, concurrency is to make sure multiple processes (programs or sub-programs) running together without influencing the others' status and operations, as if they are running simultaneously.

Some researchers like Andrews perceive concurrency simply as "two or more processes that work together to perform a task", and classify parallelism as a subset of concurrency [1]. Not only did they fail to recognize the case that two concurrent processes could perform different tasks (such as playing music and browsing websites), but also they are unable to identify that both operation level and micro-operation level parallel processing do not fall in the scope of concurrency. In other words, neither concurrency nor parallelism is a superset of the other, but they have overlaps. Moreover, there are some researchers perceive multi-tasking as a subset of multi-threading [11]; while some other researchers equate the multi-tasking with multi-threading. These are not significant issues as long as they are clearly qualified in the context.

2.3 Distributed System

The third feature of cloud is distributed system. This concept itself is controversial and various researchers differ in defining a distributed system. According to Mullender, a distributed system refers to several computers cooperating with each other to carry out task(s) together, and it has three primary features: Multiple computers, Interconnection and Shared State [10]. Tel defines a distributed system as "an interconnected collection of autonomous computers, processes or processors" [18]. In his stance, the scope of a node in a distributed system extends to include process and processor, rather than merely computer. He uses the term "autonomous" to characterize the nodes, indicating that each node shall have its own private control. This idea has much overlaps with parallel processing discussed above, as he classifies MISD and MIMD architectures into distributed system, leaving only the SIMD model out of the scope. However, apart from "processor", his extension to cover "process" is not widely accepted. Santoro argues that a distributed computing environment is "a finite collection of computational entities" communicating by message passing, to carry out collective objectives, where each entity is capable of storing and processing data, transmitting messages, setting alarm clock and changing status [13]. This explicitly introduces data-storage to qualify nodes. Some researchers also treat multi-processor computer as "distributed computer" [8].

Since so many researchers have largely different views towards distributed system, Enslow tries to make a clarification. According to [5], he concludes that a distributed system contains five fundamental components, which are:

Multiple general-purpose processing resources – "Resource" includes not only computers, but also other equipments whichever are capable of conducting generic computation – for example, the smart phones or PDAs.

Physical distribution and interconnection – Interconnection between the processing components shall be realized only through network, thus parallel computing with shared memory (SIMD and MIMD) is an exception.

Unifying of system operation – Enslow insists on the existence of a high-level OS to unify the computations of individual processing components, though he stresses that there should not be a significant hierarchy between this high-level OS and local OSes with respect to local physical resources.

Transparency – This was originally suggested by Tanenbaum in [17], who referred to a situation that the system hides the existence of multiple computers from the user, thus from the user's view, the whole system acts like a single virtual computer. Enslow extends this idea by also suggesting that the user should be communicating with the high-level OS, which represents the system as a whole, rather than to any local OS.

Cooperative autonomy – the networked interactions among the nodes comply with a "two-party cooperative protocol", where on the basis of cooperation, every unit reserves the right to refuse a computational request, realizing a "cooperative autonomy". For that reason, master-slave model does not fall in the scope of distributed system.

This paper generally agrees with Enslow's perspective except for the following three points. The first point is about the necessity of the alleged high-level OS. In fact, the concept of OS suggested here is different from the normal OS like Windows or UNIX. It is more likely to be a set of well-defined policies as an addition to the various OSes used by the distributed nodes, to let all of them cooperatively build up an abstract OS, in order to unify the different system operations among the nodes. In that sense, a specifically developed program or sub-program that could effectively handle the interactions between different OSes could well substitute the existence of such a high-level OS. Second, Enslow's generalization on "resource" is incomplete. Particularly, he fails to take into account the data storage and supply issues pointed out by Santoro. A specific component that does only store and supply relevant data without joining in processing should also be included in this scope. Today, the computing resources usually refer to computers or like things, which inherently have data storage hardware. However, in the future, it might be possible to separate processing and data storage into different devices at different places, in which case one might have a processor in New York and a hard disk in London to make up his/her "computer". In that sense, the hard disk in London should also be included in the distributed system. In other words, the description should be extended to "multiple general-purpose processing resources and data-provisioning resources". Furthermore, although cooperative processing in distributed systems (usually referred to as distributed computing) has been introduced in his theory, Enslow and many other researchers fail to identify the limitation of its level – distributed computing is only possible to achieve parallel processing at program and sub-program level, but not possible at the statement level, operation level or micro-operation level. It could be treated as a special form or a subset of parallelism.

A distributed system allows wide-geographical-range sharing of information and resource, and allows a collection of lower-end cost-effective computers to "merge" to build up a powerful "supercomputer" while keeps their autonomous status. Furthermore, with relevant data-duplication mechanisms, the reliability of a distributed system is much higher than a single computer. Nonetheless, a distributed system would encounter four major issues: Unit Failure, Unreliable Communication, Insecure Communication, and Costly Communication [10]. All of these bring much higher maintenance cost than a centralized system like a single computer. In addition, the size of a cloud is based on the size of its inherent distributed system. With respect to the network range of the inherent distributed systems, cloud can largely be classified into three categories: local area cloud, metropolitan area cloud and wide area cloud, and smaller-sized cloud could build up larger-sized cloud. Based on its public accessibility, a cloud can also be characterized as a private cloud or a public cloud.

2.4 Client-Server Paradigm

The client-server model is the most fundamental model in network technologies, and there is no need to discuss its popularity. Ince has conducted an extensive research specializing client-server model in distributed systems and suggests five major benefits, i.e. Openness, Scalability, Specialization, Reliability and Design flexibility [7]. Despite the benefits, the emphasis of client-server paradigm here is between the real customers requiring services and the cloud server. A cloud server is an integrated cluster of computing and data-provisioning resources, which employs all the above technologies. This powerful cluster is regarded as a single virtual entity due to its transparent property. A typical use case is that a customer links to the cloud server requesting for specific services, and then the cloud analyses the tasks, divides them into pieces of computational sub-tasks and data requests, and distributes them to the corresponding cloud units for further processing. For that reason, in a large-sized cloud, although its existence is not essential, the high-level OS suggested by Enslow is highly recommended, in order to cope with the complexity of coordination.

At the advanced stage, thin client-server model shall be employed. This is a specific branch of client-server paradigm, where the client depends heavily on the server, which takes over the computational role of the client's processor. In contrast to the traditional fat client that performs computations locally, thin client does not involve or involves in very limited degree of data-processing [14]. At the ultimate stage, the devices that users hold are no longer computers, but terminals, which only provide GUI and input devices. A user works with input devices, but every calculation is conducted, and every piece of data is saved in those computers hidden somewhere in the Internet that s/he does not even know. The only thing s/he knows is that the desired result is worked out by the cloud and displayed on the screen. In that sense, households will only use these thin terminal devices with network access to the cloud nexus, to benefit from inexhaustible supply of utility computing.

3. BUSINESS PERSPECTIVE

3.1 Emergence of SaaS, PaaS and IaaS

In recent years, three concepts have been increasingly popular: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The three XaaS build up the three layers of cloud businesses. However, very few people have considered why it is divided into these three layers. In fact, the philosophy underneath derives from the theories of economics. Economists usually divide an industry into five stages with five value-adding activities [3], i.e.:

A1. Gathering of raw materials

A2.Transporting and storing

A3. Manufacturing of components

A4. Assembling

A5. Distribution of end products

This is referred to as "production chain".

However, the production chain in cloud computing industry is very different. First of all, the transportation and distribution are all via the Internet, with which efficiency is extremely high. This, along with the dramatic decrease of disk space cost, makes A2 and A5 insignificant, eliminating the gaps between A1, A3; whereas the occupation and access to the Internet is the key. Thereby provisioning of basic infrastructure, which involves the purchasing of computer & datacenter clusters, settlement of networking media in order to meet the cloud hardware requirements, becomes a significant value-adding activity. Secondly, the end products are the services provided by miscellaneous software applications while the raw materials and/or semi-developed components are whatever could be used to develop the required software. The wide sources of raw materials and components, and the elimination of the gap between A1 and A2, merge A1-A3 into a single activity – software development. Finally, a platform (or operating system) is a special type of software that does not provide services for end users but play an intermediate role, linking serviceable software with hardware [16]. This special existence in this industry creates another business opportunity. In other words, the special value-adding activities in the cloud computing industry could be divided into three layers, starting from the provisioning of basic infrastructure and software platform to the end products. The innovative idea of employing pay-as-you-go mechanism in the consumption of infrastructure, platform and software results in the three business forms of XaaS.

3.2 Software as a Service

SaaS has been widely discussed by researchers, and it is now the most practical means. Traditionally, a customer buys a software program for life-long use, with a perpetual license or by other forms. The software program is treated as a product. SaaS overthrows this ideology, and allows customers to buy the software program only for the time they use. This pay-as-you-go model regards software programs as services. Cloud computing "converts capital expenses to operating expenses" [2]. Dym has analyzed the benefits of SaaS approach from both customers and software providers' stances [4]. However, the items he has summarized have much overlaps, and he also fails to identify some important underlying reasons. Therefore the benefits are re-summarized as follows:

3.2.1 Customer benefits

Lower cost of ownership – Since now customers only pay for the time that the software is used, the reduction of cost is the most obvious result.

Lower cost in problem solving – The cost includes monetary cost, time consumed, and the loss caused by software failure and the prior investment in technical expertise. In SaaS, the service providers take over the jobs of technical problem solving.

Ubiquitous access – With the Internet, customers could access to the services nearly everywhere. When the wireless network technologies develop into an advanced stage, covering as wide as mobile phone signals, cloud computing will become really ubiquitous.

Wider range of choices – The SaaS model enables customers to pay a small amount to "taste" the program, and switch to another later. Therefore customers could have much more choices and better use experience of various software applications to form their software track.

3.2.2 Vendor Benefits

Larger market – Since the cost of software is lower, much more customers now could buy what they could not afford before, resulting in a much larger market.

Improved customer relationship – As customers do not need to struggle on technical problems, they will complain much less.

More stable monthly revenue inflow – The pay-as-you-go nature spreads the revenue of software programs, resulting in more predictable monthly revenue inflow, which in turn enables the company to better develop financing strategies.

Lower entry barrier – Dym suggests that SaaS creates better opportunities for new comers to challenge the incumbents. However, he fails to identify the underlying reasons, where it is the relative ease of distribution, marketing and financing that collectively lower the entry barrier.

Better positioning – The famous business strategist Michael Porter has suggested the Three Generic Strategies Framework, which generalizes a firm's strategies into three categories, respectively differentiation, low price and focusing on particular groups, especially in local areas [12]. Although this paper does not extend to criticize this business model, the third strategy is much less applicable in cloud industry as the Internet has made geographic distance insignificant. In other words, companies now consider only two choices: either to differentiate, or to lower the price. This helps managers better develop positioning strategies.

Shorter software development life cycle (SDLC) – Traditionally, testing and debugging account for a large proportion of the SDLC. Researchers like Dym suggest that SaaS manufacturers could perform smaller amount of testing as they can use agile SDLC processes. However, they fail to identify the reasons underneath. The first thing they neglect is the cultural impact of SaaS, where customers are allowed to and tend to pay small amounts to "taste" some key functions of applications before they are fully-developed, becoming more comfortable to rapid changes. Moreover, many researchers fail to identify the possible impacts brought by the development of IaaS and PaaS in the future. The cost of testing is higher if the infrastructure and platform lacks standardization, vice versa. As IaaS develops, hardware infrastructure will be more standardized, thus reducing the need to do testing in different hardware equipment. In the development of electricity, after the voltage standard for household use was developed, for example 230V, manufacturers do not need to test the products under 150V, 180V or any other voltage levels. The impact of PaaS is not easy to predict, as whether or not platform is going to be standardized is inconclusive (will be discussed later). Nonetheless, the impact of PaaS is much less significant than that of IaaS.

More cost-effective in marketing – Dym suggests that customers' habits have shifted from "an outbound world driven by field sales and print advertising to an inbound world driven by Internet search". This is actually another aspect of the cultural impact. SaaS manufacturers are positioning ideally in accordance with this shift, and SaaS marketing is very cost-effective.

3.3 Platform as a Service & Infrastructure as a Service

Research on PaaS and IaaS is very limited. The basic idea is also to employ the pay-as-you-go mechanism. However, the market situations are very different from that of SaaS. According to theories of economics [9], the PaaS and IaaS markets are oligopolistic ones, rather than a monopolistic competitive one. The former refers to the situation that there are only several tycoons such as Microsoft and Apple dominating the market, while the latter is very much similar to free market in commonsense (In theories of economics, the term "free market" has a different meaning and refers to an ideal market situation). Oligopolistic market and monopolistic competitive market are very different, and this might be one of the reasons that there are much less research has been carried out on PaaS and IaaS.

In the case of PaaS, it is difficult to predict the future situation, platform could either be diversified or standardized. Normally, end users favor standardization of hardware but diversification of software. Unfortunately, OS is the intermediate between hardware and software, and some people prefer standardization while others prefer diversification. Today, lots of users have preferences on Mac OS, Ubuntu and so forth, though Windows remains at dominant position. Whether users will prefer diversified or standard OSes is inconclusive. The research on PaaS is far from adequate.

For IaaS, economies of scale (EoS) and standardization are the key to success. Economies of scale refer to the property where average-total-cost of products falls as quantity increases [9]. For example, after purchasing a production line, an extra product will spread a percentage of this cost, driving down the average cost. In the provisioning of hardware infrastructure, people care much less about differentiation and prefer the opposite. Thereby dominant power will probably occur at last as the suppliers with higher market share will have significant competitive advantage over others, eventually making IaaS a monopoly market.

Empirical evidences have proven this. In [2], Armbrust et. al. point out the cost of setting up a mid-size datacenter (hundreds or thousands of computers) is between five times and seven times of that for a large datacenter, as shown in Table 1. They also suggest that some cloud datacenters have been built in some unexpected locations, including *"Quincy, Washington and San Antonio, Texas"*. The reason is the costs of electricity supply and cooling, which amounts to one third of the total costs, are relatively lower in those areas, as shown in Table 2.

Table 1. EoS of Datacenters: Medium-size VS Large-size

Technology	Cost in Medium DC	Cost in Very Large DC	Ratio
Network	$95 per Mbit/sec/month	$13 per Mbit/sec/month	7.1
Storage	$2.20 per GB/month	$0.40 per GB/month	5.7
Administration	≈ 140 Servers / Administrator	>1000 Servers / Administrator	7.1

Table 2. Region Difference in Electricity Price

Price per KWH	Where	Possible Reasons Why
3.6¢	Idaho	Hydroelectric power; not sent long distance
10.0¢	California	Electricity transmitted long distance over the grid; limited transmission lines in Bay Area; no coal fired electricity allowed in California.
18.0¢	Hawaii	Must ship fuel to generate electricity

3.4 Utility Computing

Many researchers believe that computing will become a type of utility, and its development will resemble that of electricity. Indeed, they have very much similarity. In the beginning, the supply of electricity was provided by individual power generators, which are only capable of supplying very limited areas. Likely, PCs are only for households. After Tesla invented the AC power model and relevant transmission methods, large-scale remote supply became possible. To exploit the economies of the scale, electricity suppliers integrated the settlement of power plants and set up electricity plants, similar to the concept of cloud. As shown in the following table:

Table 3. Utility Comparison: Electricity VS Computing

Time Period	Utility Type	
	Electricity	***Computing***
Beginning	Individual power generators	Individual PCs
Development	AC power model	Cloud computing model
Media	Extra-high-voltage transmission	Internet
Economies of Scale	Integrated clusters of power plants	Integrated clusters of computers

Evidences have shown the power of EoS and the possibility of supplying computing utility with massive production, partially demonstrated this trend.

Furthermore, most researchers have not recognized the ultimate stage of utility computing. When access to electricity finally become very easy and reliable and electrical devices become highly developed, households do not need to install individual power generators at home. Similarly, when the supply of utility computing becomes cheap and reliable, when the access to the cloud becomes ubiquitous, when its transmission becomes extremely fast, and when client terminal devices become highly developed, the times of PC will fade out. At that new era, households use terminals, which only provide GUI and input devices, with wired or wireless connection to the cloud nexus, to access various resources. This is highly possible due to the powerfulness of EoS, learning curve economies [9] and the cultural impact brought by XaaS, as well as the increasing practices of thin client-server computing. The bottleneck is the networking technology: it is whether the delay of remote computation could be low enough to emulate PCs performing computation locally that determines whether this stage will be realized. In fact, the idea of developing terminal devices capable of connecting to a data nexus in wireless form at any point of the earth, is similar to the proposal of ubiquitous wireless energy transfer, the ultimate stage of the remote energy transfer paradigm, suggested nearly a century ago by the famous physicist and electrical engineer – Nikola Tesla.

4. OTHER RESEARCHERS' PERSPECTIVE

4.1 Sun Microsystems Perspective [15]

Sun Microsystems is one of the main supporters who believes that cloud computing would be the next generation of network computing. *"Network is the Computer"* is the phrase they coined to demonstrate their point of view towards cloud computing. They take a comprehensive stance to different types of clouds and different cloud services to the extent that the corresponding practice facilitates the deployment of application and innovation. Furthermore, they suggest four aspects of cloud computing:

"Virtualization as the standard deployment means" **–** Virtualization abstracts the hardware, enabling software not to be deployed with specific physical devices, resulting in a significantly enhanced flexibility of software deployment. It includes two parts: virtual machine and virtual appliances.

"On-demand, self-service, pay-by-use model incorporated" **–** From the views of enterprises, the on-demand model facilitates service objectives in their performance and capacity; the self-service nature enables organizations to flexibly control the workload and other target parameters; and the pay-by-use model could assure a minimum level of service, preventing waste of resources.

"Applications are composed and are built to be composable" **–** As a consequence of the self-service, pay-by-use model, software development could easily outsource third-party or open-source software components rather than over-rely on self-development. Similarly, those components would incorporate a composable nature to widen its use.

"Programmable infrastructure" **–** Due to its virtualization nature, hardware infrastructure also becomes virtual, and becomes programmable. Traditionally, system architects take the responsibility to specify the way a set of hardware equipments to interrelate with each other and function collaboratively. With cloud computing, which makes creating a virtual machine become as easy as creating a new thread for a program, developers will take over the role of architects.

Generally, Sun's understanding is limited on the SaaS layer. The alleged virtualization aspect is actually the very nature of web-based applications. The second point is another way to describe the pay-as-you-go approach. The third aspect is actually the result of specializing production chain in software development, and splits it into A3 and A4. The fourth aspect is a notable one. In pure cloud computing model, the cloud will act as a single entity, thus there is no need to consider and deploy the interconnections between different computers and hardware devices. Developers could call for additional virtual processors whenever they need. This implies the vanishment of system architects.

4.2 IBM Perspective [6]

IBM perceives cloud computing as a service acquisition and delivery model for IT resources with two aspects:

"From a user perspective, cloud computing provides a means of acquiring computing services without requiring understanding of the underlying technology. From an organizational perspective, cloud computing delivers services for consumer and business needs in a simplified way, providing unbounded scale and differentiated quality of service to foster rapid innovation and decision making".

IBM identifies four technology attributes of cloud computing:

"Services focused" **–** Cloud computing is about providing services to users from all over the world with internet access. Therefore it must be built on a service-oriented architecture (SOA)

"Shared, highly scalable, networked infrastructure" **–** This indicates a massive scalability of standardized, highly efficient, shared, virtualized computing and data-supplying resources.

"Automated service delivery" **–** With cloud computing, service management is request-focused with negligible marginal labor costs. It dynamically optimizes the workloads and data across the shared resource with little interference by the service providers. When a particular resource is released, customers immediately gain access to it.

Enhanced, standardized user experience" **–** It provides the clients with user-friendly interfaces with easy access to the desired information.

Similar to Sun, IBM's understanding is also limited at the SaaS level, which seems to be a common limitation of enterprise practitioners. The difference is that Sun stresses on the developers' perspective while IBM focuses on end users' perspective. Actually, IBM's famous SOA approach has much overlaps with SaaS; the automated service delivery is the basic nature of Internet; and the enhanced user experience indicates requirements for lower technical expertise. However, IBM misunderstands cloud computing to the extent that cloud services are standardized. In fact, cloud services will become highly diversified as discussed above. Furthermore, IBM's understanding on infrastructure is limited to the extent that it is shared, scalable and networked. They not only fail to identify that it will finally becomes a type of service, but also lacks the in depth technical insight as Sun possesses.

4.3 Berkeley Perspective [2]

Armbrust et. al. suggest that:

"Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS). The datacenter hardware and software is what we will call a Cloud. When a Cloud is made available in a pay-as-you-go manner to the general public, we call it a Public Cloud; the service being sold is Utility Computing. We use the term Private Cloud to refer to internal datacenters of a business or other organization, not made available to the general public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not include Private Clouds. People can be users or providers of SaaS, or users or providers of Utility Computing."

Indeed, scholars have much deeper understanding than enterprises. The concept of cloud computing suggested here covers all the three layers of service, system software and hardware. However, they narrowly perceive cloud computing merely as a commercial service and utility, excluding the private clouds. Applying induction into another utility: if a government electricity station does not provide electricity to public society, what it generates is not electricity and this station is not an electricity station. Apparently this is a ridiculous generalization.

Furthermore, they suggest two additional benefits. First, another possible reason to become a cloud provider is to *"leverage existing investment"*, where they demonstrate with the example of Amazon Web Services. Second, traditional software provider might offer cloud options as a means to defend their market share when its competitors switch to cloud computing. Moreover, they emphasize the economic benefits brought by the elasticity of cloud computing, where XaaS models offer transferability to lower risks. In theories of economics [9], this is referred to as lowering opportunity cost.

5. CONCLUSION

Cloud computing is a very complicated concept and has been abused and misused to a significant extent, causing confusions and controversies, particularly by the enterprises that launch practices. Cloud computing involves both computing technologies and business ideologies. The technological perspective of cloud computing involves four major features. Parallel processing in theory enables multiple computing resources in the cloud to speed up the calculation by n times, where n stands for the number of processors contained. Concurrency enables multiple tasks and processes to be carried out in an overlapped time, without influencing each other. A distributed system contains multiple physically distributed general-purpose computing and data-provisioning resources, interconnected by network, and operating under the philosophy of cooperative autonomous, unifying operations and collectively appearing as a single powerful entity. Client-server, particularly the thin client-server model, separates computation and GUI, abstracting away the complexity from the users' stances and enabling them to access to a powerful "supercomputer".

The business perspective derives from the economic concept of production chain and the introduction of pay-as-you-go mechanism in this industry. Based on its special value-adding activities, the cloud business operations are divided into SaaS, PaaS and IaaS, among which, SaaS is most widely practiced with abundant research to support. Furthermore, it is hypothesized that PC will gradually disappear, and instead, the substitute terminal devices will provide only GUI, input devices and network access, deriving utility computing from the cloud. Moreover, enterprise practitioners usually equate it with SaaS, under-perceiving the magnificence of cloud computing, while scholars generally have deeper understanding.

6. FUTURE WORK

The research on PaaS is far from adequate. This is perhaps because OS is a special existence that lay between hardware and software, thus containing features of both markets. Another reason may be that the market of PaaS is an oligopolistic one, much different from the usual markets. IaaS also suffers from limited research. Both PaaS and IaaS need further investigation and analysis.

7. ACKNOWLEDGMENTS

First of all, I want to thank my mentor Dr. Peter Blanchfield, for his patient guidance on me over the past year. Without his assistance, I would not have achieved so much. Secondly, I would like to thank my family, for supporting my undergraduate study. Finally, I want deliver my wishes to all my friends, who drive away my uneasiness and loneliness during my university life in a foreign country.

REFERENCES

[1] Andrews, G. R. (2000) Foundations of Multithreaded, Parallel and Distributed Programming, Addison Wesley.

[2] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D. A., Rabkin, A., Stoica, I. & Zaharia, M (2009) 'Above the Clouds: A Berkeley View of Cloud Computing', Technical Report No. UCB/EECS-2009-28. DOI = http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html

[3] Brickley, J. A., Smith, C. W. & Zimmerman, J. L. (2007) Managerial Economics & Organizational Architecture, 4th edition, McGraw-Hill/Irwin.

[4] Dym R. (2010) 'Why Software as a Service: Helping Our Customers Reduce Costs and Increase Revenue', OpSource White Paper. DOI= http://www.opsource.net/saas/wp_why_saas.pdf

[5] Enslow, P. H. (1978) 'What is a "Distributed" Data Processing System?', Computer, January, vol. 11, no. 1, pp. 13-21.

[6] IBM (2008), IBM Perspective on Cloud Computing, IBM Corporation

[7] Ince, D. (2002) Developing Distributed and E-Commerce Applications, Pearson Education Limited.

[8] Jensen, E. D. (1978) 'The Honeywell Experimental Distributed Processor', Computer, vol. 11, no .1, pp. 28-38.

[9] Mankiw, N. G. (2007) Principles of Microeconomics, 4th edition, Thomson South-Western.

[10] Mullender, S. (1994) Distributed Systems, 2nd edition, Addison-Wesley

[11] National Instruments (2008) 'Differences Between Multithreading and Multitasking for Programmers', Multicore Programming Fundamentals Whitepaper Series. DOI= http://zone.ni.com/devzone/cda/tut/p/id/6424

[12] Porter, M. (1980) Competitive Strategy, New York: Free Press.

[13] Santoro, N. (2007) Design and Analysis of Distributed Algorithms, John Wiley & Son Inc.

[14] Shapiro, J., Boyce, J., Policht, M., Patterson, B. & Leathers, S. (2003) Windows Server 2003 Bible, Wiley Publishing, Inc.

[15] Sun Microsystems (2009), 'Introduction to Cloud Computing Architecture', White Paper, 1st edition, Sun Microsystems Inc.

[16] Tanenbaum, A. S. (2009) Modern Operating Systems, 3rd edition, Pearson Education, Inc.

[17] Tanenbaum, A. S. (1989) Computer Networks, 2nd edition, Prenticce-Hall.

[18] Tel, G. (2000) Introduction to Distributed Algorithms, 2nd edition, Cambridge University Press.

[19] Xavier, C. & Iyengar, S. S. (1998) Introduction to Parallel Algorithms, John Wiley & Sons