Cloud Computing: Paradigms and Technologies

Ahmed Shawish and Maria Salama

Abstract Cloud Computing has recently emerged as a compelling paradigm for managing and delivering services over the internet. It is rapidly changing the landscape of information technology, and ultimately turning the long-held promise of utility computing into a reality. With such speedy progressing and emerging, it becomes crucial to understand all aspects about this technology. This chapter provides a comprehensive overview on the Cloud's anatomy, definition, characteristic, affects, architecture, and core technology. It clearly classifies the Cloud's deployment and service models, providing a full description of the Cloud services vendors. The chapter also addresses the customer-related aspects such as the Service Level Agreement, service cost, and security issues. Finally, it covers detailed comparisons between the Cloud Computing paradigm and other existing ones in addition to its significant challenges. By that, the chapter provides a complete overview on the Cloud Computing and paves the way for further research in this area.

1 Introduction

Cloud Computing has recently emerged as a compelling paradigm for managing and delivering services over the internet. The rise of Cloud Computing is rapidly changing the landscape of information technology, and ultimately turning the

A. Shawish (✉)
Department of Scientific Computing, Faculty of Computer and Information Science,
Ain Shams University, Cairo, Egypt
e-mail: ahmed.gawish@fcis.asu.edu.eg

M. Salama
Department of Computer Science, Faculty of Informatics and Computer Science,
British University in Egypt, Cairo, Egypt
e-mail: maria.salama@bue.edu.eg

F. Xhafa and N. Bessis (eds.), *Inter-cooperative Collective Intelligence: Techniques and Applications*, Studies in Computational Intelligence 495,
DOI: 10.1007/978-3-642-35016-0_2,

long-held promise of utility computing into a reality. The latest emergence of Cloud Computing is a significant step towards realizing this utility computing model since it is heavily driven by industry vendors. It attracts business owners due to its ability to eliminate the provisioning plan overhead, and allows enterprises to start from the small scale and dynamically increase their resources simultaneously with the increase of their service demand. Cloud Computing promises to deliver reliable services through next-generation data centers built on virtualized compute and storage technologies. Users will be able to access applications and data from a Cloud anywhere in the world following the pay-as-you-go financial model.

With such speedy progressing of the Cloud Computing and emerging in most of the enterprise business and scientific research areas, it becomes crucial to understand all aspects about this technology. The aim of this chapter is to provide a complete overview on the Cloud Computing through a comprehensive descriptions and discussion of all aspects of this technology. In this chapter, Cloud Computing anatomy is presented along with the essential definitions from different perspectives and prominent characteristics. The affects of Cloud Computing on organizations and enterprises were also addressed in terms of time, management and operational costs. The architecture design of Clouds has been discussed, as well as the key technologies behind it; such as virtualization and web services.

Clouds are classified according to their deployment models as private, community, public and hybrid Clouds. Clouds also offer different service models; software, platform and infrastructure as service. In such perspective, the chapter addresses the Cloud Computing services classification by clearly differentiating between deployment and service models in one hand, and beneath each of them on the other hand. The Cloud providers and vendors are also addressed and described.

As more providers offer computing services using Cloud infrastructure, the method of determining the right price for users become crucial for both providers and consumers. The Service Level Agreement that drives the relation between the provider and the consumer becomes also of a great significance. Similarly the security of the data across the Cloud is gaining a great interest due to its sensitivity. This chapter discusses the Cloud customer-related aspects in terms of the Service Level Agreements, service cost, service pricing, and security issue.

In fact, the Cloud Computing is built on top of several other technologies, for example Distributed Computing, Grid Computing, and Utility Computing. This chapter covers a comprehensive comparison between the Cloud Computing and preceding paradigms; in terms of architecture, resources sharing, QoS guarantees and security.

The challenges facing the new paradigm; such as security, availability and resources management; should be carefully considered in future research in order to guarantee the long-term success of Cloud Computing. The chapter tackles these challenges and paves the way for further research in this area.

2 Anatomy of Cloud Computing

This section presents a general overview of Cloud Computing; including its definitions, characteristics, and organizational affects. Clouds architecture is also addressed, as well as the key technologies on which Cloud Computing rely.

Cloud Computing is the new cost-efficient computing paradigm in which information and computer power can be accessed from a web browser by customers. Cloud Computing is the Internet-based development and use of computer technology. Loosely speaking, Cloud Computing is a style of computing paradigm in which typically real-time scalable resources such as files, data, programs, hardware, and third party services can be accessible from a Web browser via the Internet to users. These customers pay only for the used computer resources and services by means of customized Service Level Agreement (SLA), having no knowledge of how a service provider uses an underlying computer technological infrastructure to support them. The service load in Cloud Computing is dynamically changed upon end-users' service requests [1].

Cloud Computing shifts the computation from local, individual devices to distributed, virtual, and scalable resources, thereby enabling end-users to utilize the computation, storage, and other application resources, which forms the Cloud, on-demand [2].

2.1 Definition

Cloud Computing has been coined as an umbrella term to describe a category of sophisticated on-demand computing services initially offered by commercial providers. It denotes a model on which a computing infrastructure is viewed as a "Cloud", from which businesses and individuals access applications from anywhere in the world on demand [3]. The main principle behind this model is offering computing, storage, and software "as a service". There are many definitions of Cloud Computing, but they all seem to focus on just certain aspects of the technology [4].

An early definition of Cloud Computing has been proposed as follows: A computing Cloud is a set of network enabled services, providing scalable, Quality of Service (QoS) guaranteed, normally personalized, inexpensive computing platforms on demand, which could be accessed in a simple and pervasive way [5].

Markus Klems claims that immediate scalability and resources usage optimization are key elements for the Cloud. These are provided by increased monitoring, and automation of resources management in a dynamic environment. Other authors disagree that this is a requirement for an infrastructure to be considered as a Cloud [6].

According to a 2008 paper by Carl Hewitt called "ORGs for Scalable, Robust, Privacy-Friendly Client Cloud Computing" published by IEEE Internet Computing, "Cloud Computing is a paradigm in which information is permanently stored in servers on the internet and cached temporarily on clients that include desktops, entertainment centers, table computers, notebooks, wall computers, handhelds, sensors, monitors, etc." [7].

A Berkeley Report in February 2009 states "Cloud Computing, the long-held dream of computing as a utility has the potential to transform a large part of the IT industry, making software even more attractive as a service" [8].

From an economic perspective, Cloud Computing is defined as follows: "Building on compute and storage virtualization technologies, and leveraging the modern Web, Cloud Computing provides scalable and affordable compute utilities as on-demand services with variable pricing schemes, enabling a new consumer mass market" [9].

From the Quality of Service perspective, Clouds have been defined as a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLA's" [10]. The definition refers to a pay-per-use economic model taken from the paradigm of Utility Computing.

Some authors focus on the business model (collaboration and pay-as-you-go) and the reduction in capital expenditure by the realization of Utility Computing. Until recently, it was often confused with the Cloud itself, but it seems now agreed that it is just an element of the Cloud related to the business model. Another major principle for the Cloud is user-friendliness [6]. Buyya et al. [11] added that to reach commercial mainstream it is necessary to strengthen the role of SLAs between the service providers and the consumers of that service. McFedries [12] described the data center (conceived as a huge collection of clusters) as the basic unit of the Cloud offering huge amounts of computing power and storage by using spare resources. The role of virtualization in Clouds is also emphasized by identifying it as a key component [11]. Moreover, Clouds have been defined just as virtualized hardware and software plus the previous monitoring and provisioning technologies [6].

The US National Institute of Standards and Technology (NIST) has published a working definition [13] that seems to have captured the commonly agreed aspects of Cloud Computing. This definition describes Cloud Computing using:

- Five characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.
- Four deployment models: private Clouds, community Clouds, public Clouds, and hybrid Clouds.
- Three service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

An encompassing definition of the Cloud taking into account Cloud features has been proposed as follows. Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs. On the other hand, looking for the minimum common denominator would lead us to no definition as no single feature is proposed by all definitions. The set of features that most closely resemble this minimum definition would be scalability, pay-per-use utility model and virtualization [14].

Obviously, the Cloud concept is still changing and these definitions show how the Cloud is conceived today.

2.2 Characteristics

The features of Cloud Computing are that it offers enormous amounts of power in terms of computing and storage while offering improved scalability and elasticity. Moreover, with efficiency and economics of scale, Cloud Computing services are becoming not only a cheaper solution but a much greener one to build and deploy IT services [15].

The Cloud Computing distinguishes itself from other computing paradigms in the following aspects [5]:

- *On-demand service*: Computing Clouds provide resources and services for users on demand. Users can customize and personalize their computing environments later on, for example, software installation, network configuration, as users usually own administrative privileges.
- *QoS guaranteed offer*: The computing environments provided by computing Clouds can guarantee QoS for users, e.g., hardware performance like CPU speed, I/O bandwidth and memory size. The computing Cloud renders QoS in general by processing Service Level Agrement (SLA) with users.
- *Autonomous System*: The computing Cloud is an autonomous system and it is managed transparently to users. Hardware, software and data inside Clouds can be automatically reconfigured, orchestrated and consolidated to present a single platform image, finally rendered to users.
- *Scalability*: The scalability and flexibility are the most important features that drive the emergence of the Cloud Computing. Cloud services and computing platforms offered by computing Clouds could be scaled across various concerns; such as geographical locations, hardware performance, and software configurations. The computing platforms should be flexible to adapt to various requirements of a potentially large number of users.

2.3 Affects

Recently, there has been a great deal of hype about Cloud Computing. Cloud Computing is on the top of Gartner's list of the ten most disruptive technologies of the next years. Since Cloud Computing makes several promises, in terms of time , management and operational costs, enterprises need to understand the affects of Cloud Computing which are focused on the following specific topics [16]:

- *The organizational change brought about with Cloud Computing*: The type of organizational change that Cloud Computing results in can be demonstrated by taking a look at, for example, IT procurement within enterprise. Simplistically, procurement is based on obtaining estimates for things, then getting those estimates signed-off by management to allow the procurement to proceed. Capital and operational budgets are kept separate in this process, and it can take several months between the decision to procure hardware and the hardware being delivered, setup and ready to use. The use of Cloud Computing can greatly reduce this time period, but there is more significant change related to the empowerment of users and the diffusion of the IT department's authority. For example, a company's training coordinator who requires a few servers to run a week-long web-based training course can bypass their IT department and run the training course in the Cloud. They could pay their Cloud usage-bill using their personal credit card and charge back the amount as expenses to their employee.
- *The economic and organizational implications of the utility billing model in Cloud Computing*: New Cloud Computing pricing models based on market mechanisms are starting to emerge, but it is not yet clear how such models can be effectively used by enterprise. An example of such models is used by Amazon's Spot Instances, which allows users to bid for unused capacity in Amazon's data centers. Amazon runs the user's instances as long as the bid price is higher than the spot price, which is set by Amazon based on their data center utilization.
- *The security, legal and privacy issues that Cloud Computing raises*: Security, legal and privacy issues are widely acknowledged as being important in Cloud Computing. Most of the security and privacy issues in Cloud Computing are caused by users' lack of control over the physical infrastructure. This leads to legal issues that are affected by a Cloud's physical location, which determines its jurisdiction. Furthermore, multi-tenancy brings the need for new solutions towards security and privacy. For example, Denial of Service (DoS)[1] attacks were a common concern even before Cloud Computing became popular, but when an application is targeted by a DoS attack in the Cloud, the user or owner could actually end-up paying for the attack through their increased resource usage. This could be significantly higher than the peak usage of that application in an in-house data-center with limited resources.

---

[1] A Denial of Service attack (DoS attack) is an attempt to make a machine or network resource unavailable to its intended users.

2.4 Architecture

Many organizations and researchers have defined the architecture for Cloud Computing. Basically, the whole system can be divided into the core stack and the management. In the core stack, there are three layers: (1) Resource (2) Platform and (3) Application.

The resource layer is the infrastructure layer which is composed of physical and virtualized computing, storage and networking resources.

The platform layer is the most complex part which could be divided into many sub-layers; e.g. a computing framework manages the transaction dispatching and/or task scheduling. A storage sub-layer provides unlimited storage and caching capability.

The application server and other components support the same general application logic as before with either on-demand capability or flexible management, such that no components will be the bottle neck of the whole system [17].

Based on the underlying resource and components, the application could support large and distributed transactions and management of huge volume of data. All the layers provide external service through web service or other open interfaces. Cloud Architecture is depicted in Fig. 1.

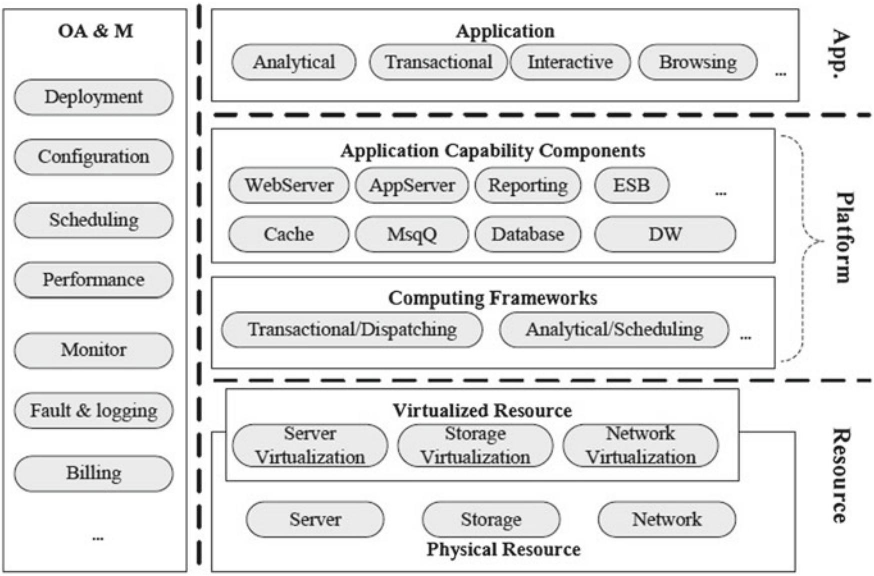

Fig. 1 Cloud architecture

2.5 Technologies behind Cloud Computing

A number of enabling technologies contribute to Cloud Computing, several state-of-the-art techniques are identified [5]:

- *Virtualization* technology: Virtualization technologies partition hardware and thus provide flexible and scalable computing platforms. Virtual machine techniques, such as VMware[2], and Xen[3] offer virtualized IT infrastructures on demand. Virtual network advances, such as Virtual Private Network[4] (VPN), support users with a customized network environment to access Cloud resources. Virtualization techniques are the bases of the Cloud Computing since they render flexible and scalable hardware services.
- *Orchestration of service flow and workflow*: Computing Clouds offer a complete set of service templates on demand, which could be composed by services inside the computing Cloud. Computing Clouds therefore should be able to automatically orchestrate services from different sources and of different types to form a service flow or a workflow transparently and dynamically for users.
- *Web service and SOA*: Computing Cloud services are normally exposed as Web services, which follow the industry standards such as WSDL[5], SOAP[6] and UDDI[7]. The services organization and orchestration inside Clouds could be managed in a Service Oriented Architecture (SOA). A set of Cloud services furthermore could be used in a SOA application environment, thus making them available on various distributed platforms and could be further accessed across the Internet.
- *Web 2.0*: Web 2.0 is an emerging technology describing the innovative trends of using World Wide Web technology and Web design that aims to enhance creativity, information sharing, collaboration and functionality of the Web. Web 2.0 applications typically include some of the following features/techniques:

---

2 VMware software provides a completely virtualized set of hardware to the guest operating system.

3 Xen is a Hypervisor providing services that allow multiple computer operating systems to execute on the same computer hardware concurrently.

4 A virtual private network (VPN) is a private computer network that interconnects remote (and often geographically separate) networks through primarily public communication infrastructures such as the Internet.

5 WSDL (Web Services Description Language) is an XML-based language that is used for describing the functionality offered by a Web service.

6 SOAP, originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of Web Services in computer networks.

7 Universal Description, Discovery and Integration (UDDI) is a platform-independent, Extensible Markup Language (XML)-based registry by which businesses worldwide can list themselves on the Internet, and a mechanism to register and locate web service applications.

– Cascading Style Sheets[8] (CSS) to separate of presentation and content
– Folksonomies (collaborative tagging, social classification, indexing and social tagging).
– Semantic Web technologies.
– REST, XML and JSON-based APIs.
– Innovative Web development techniques such as Ajax.
– XHTML and HTML markup.
– Syndication, aggregation and notification of Web data with RSS or Atom feeds.
– Mashups, merging content from different sources, client- and server-side.
– Weblog publishing tools.
– Wiki to support user-generated content.
– Tools to manage users' privacy on the Internet.

The essential idea behind Web 2.0 is to improve the interconnectivity and interactivity of Web applications. The new paradigm to develop and access Web applications enables users access the Web more easily and efficiently. Cloud Computing services in nature are Web applications which render desirable computing services on demand. It is thus a natural technical evolution that the Cloud Computing adopts the Web 2.0 technique.

• *World-wide distributed storage system*: A Cloud storage model should foresee:

  – A network storage system, which is backed by distributed storage providers (e.g., data centers), offers storage capacity for users to lease. The data storage could be migrated, merged, and managed transparently to end users for whatever data formats. Examples are Google File System and Amazon S3.
  – A distributed data system which provides data sources accessed in a semantic way. Users could locate data sources in a large distributed environment by the logical name instead of physical locations. Virtual Data System (VDS) is good reference.

• *Programming model*: Users drive into the computing Cloud with data and applications. Some Cloud programming models should be proposed for users to adapt to the Cloud infrastructure. For the simplicity and easy access of Cloud services, the Cloud programming model, however, should not be too complex or too innovative for end users. The MapReduce is a programming model and an associated implementation for processing and generating large data sets across the Google worldwide infrastructures. The MapReduce and the Hadoop are adopted by recently created international Cloud Computing project of Yahoo!, Intel and HP.

[8] CSS is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language; most commonly to style web pages written in HTML and XHTML.

2.6 Advantages of the Use of Cloud Technology

The advantages of Cloud Computing is that it offers enormous amounts of power in terms of computing and storage while offering improved scalability and elasticity. Moreover, with efficiency and economics of scale, Cloud Computing services are becoming not only a cheaper solution but a much greener one to build and deploy IT services.

The Cloud Computing distinguishes itself from other computing paradigms in the following aspects [5]:

- *On-demand service provisioning*: Computing Clouds provide resources and services for users on demand. Users can customize and personalize their computing environments later on, for example, software installation, network configuration, as users usually own administrative privileges.
- *QoS guaranteed offer*: The computing environments provided by computing Clouds can guarantee QoS for users, e.g., hardware performance like CPU speed, I/O bandwidth and memory size. The computing Cloud renders QoS in general by processing Service Level Agreement (SLA) with users.
- *Autonomous System*: The computing Cloud is an autonomous system and it is managed transparently to users. Hardware, software and data inside Clouds can be automatically reconfigured, orchestrated and consolidated to present a single platform image, finally rendered to users.
- *Scalability and flexibility*: The scalability and flexibility are the most important features that drive the emergence of the Cloud Computing. Cloud services and computing platforms offered by computing Clouds could be scaled across various concerns, such as geographical locations, hardware performance, and software configurations. The computing platforms should be flexible to adapt to various requirements of a potentially large number of users.

3 Classifying Clouds and Vendors

In this section, different classifications of Clouds are presented, as well as vendors are nominated.

There are diverse dimensions to classify Cloud Computing, two commonly used categories are: service boundary and service type [17].

- From the service boundary's view, Cloud Computing can be classified as public Cloud, private Cloud and hybrid Cloud. The public Cloud refers to services provided to external parties. The enterprises build and operate private Cloud for themselves. Hybrid Cloud shares resources between public Cloud and private Cloud by a secure network. Virtual Private Cloud (VPC) services released by Google and Amazon are examples of Hybrid Cloud.

- From the service type's view, Cloud Computing can be classified as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). SaaS provide services to end users, while IaaS and PaaS provide services to ISV and developers - leaving a margin for 3rd party application developers.

3.1 Deployment Models

The Cloud model promotes four deployment models:

- *Private Cloud*: The Cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.
- *Community Cloud*: The Cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.
- *Public Cloud*: The Cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling Cloud services.
- *Hybrid Cloud*: The Cloud infrastructure is a composition of two or more Clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., Cloud bursting for load-balancing between Clouds).

3.2 Service Models

Cloud Computing is gaining popularity to the extent that the new XaaS service category introduced will gradually take the place of many types of computational and storage resources used today [18]. Cloud Computing delivers infrastructure, platform, and software (application) as services, which are made available as subscription-based services in a pay-as-you-go model to consumers. These services in industry are respectively referred to as Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) [19]. Table 1 summarizes the nature of these categories and lists some major players in the field [20].

3.2.1 Infrastructure-as-a-Service

Infrastructure-as-a-Service, also called Hardware-as-a-Service was coined possibly in 2006. As the result of rapid advances in hardware virtualization, IT automation and usage metering& pricing, users could buy IT hardware, or even an entire data center, as a pay-as-you-go subscription service [5]. Infrastructure-as-a-Service

Table 1 Cloud computing services classification

Category	Characteristics	Product type	Vendors and products
SaaS	Customers are provided with applications that are accessible anytime and from anywhere	Web applications and services (Web 2.0)	SalesForce.com (CRM) Google documents Clarizen.com (Project management) Google mail (automation)
PaaS	Customers are provided with a platform for developing applications hosted in the Cloud	Programming APIs and frameworks; Deployment system	Google AppEngine Microsoft Azure Manjrasoft Aneka
IaaS/HaaS	Customers are provided with virtualized hardware and storage on top of which they can build their infrastructure	Virtual machines management infrastructure, Storage management	Amazon EC2 and S3; GoGrid; Nirvanix

(IaaS) or Hardware-as-a-Service (HaaS) solutions deliver IT infrastructure based on virtual or physical resources as a commodity to customers. These resources meet the end user requirements in terms of memory, CPU type and power, storage, and, in most of the cases, operating system as well. Users are billed on a pay-per-use basis. They have to set up their applications on top of these resources that are hosted and managed in data centers owned by the vendor.

Amazon is one of the major players in providing IaaS solutions. Amazon Elastic Compute Cloud (EC2) provides a large computing infrastructure and a service based on hardware virtualization. By using Amazon Web Services, users can create Amazon Machine Images (AMIs) and save them as templates from which multiple instances can be run. It is possible to run either Windows or Linux virtual machines, for which the user is charged per hour for each of the instances running. Amazon also provides storage services with the Amazon Simple Storage Service (S3), users can use Amazon S3 to host large amount of data accessible from anywhere [20].

3.2.2 Platform-as-a-Service

Platform-as-a-Service solutions provide an application or development platform in which users can create their own application that will run on the Cloud. More precisely, they provide an application framework and a set of API that can be used by developers to program or compose applications for the Cloud. PaaS solutions often integrate an IT infrastructure on top of which applications will be executed. This is the case of Google AppEngine and Microsoft Azure, while other solutions, such as Manjrasoft Aneka, are purely PaaS implementations.

Google AppEngine is a platform for developing scalable web applications that run on top of data centers maintained by Google. It defines an application model and provides a set of APIs that allow developers to take advantage of additional services such as Mail, Datastore, Memcache, and others. AppEngine manages the execution of applications and automatically scales them up/down as required. Google provides a free but limited service, while utilizes daily and per minute quotas to meter and price applications requiring a professional service.

Azure is a Cloud service operating system that serves as the development, runtime, and control environment for the Azure Services Platform. By using the Microsoft Azure SDK, developers can create services that leverage the .NET Framework. These services have to be uploaded through the Microsoft Azure portal in order to be executed on top of Windows Azure. Additional services, such as workflow execution and management, web services orchestration, and access to SQL data stores, are provided to build enterprise applications.

Aneka, commercialized by Manjrasoft, is a pure PaaS implementation and provides end users and developers with a platform for developing distributed applications for the Cloud by using .NET technology. The core value of Aneka is a service oriented runtime environment that is deployed on both physical and virtual infrastructures and allows the execution of applications developed by means of various programming models. Aneka provides a Software Development Kit (SDK) helping developers to create applications and a set of tools for setting up and deploying Clouds on Windows and Linux based systems. Aneka does not provide an IT hardware infrastructure to build computing Clouds, but system administrators can easily set up Aneka Clouds by deploying Aneka containers on clusters, data centers, desktop PCs, or even bundled within Amazon Machine Images [20].

3.2.3 Software-as-a-Service

Software or an application is hosted as a service and provided to customers across the Internet. This mode eliminates the need to install and run the application on the customer's local computers. SaaS therefore alleviates the customer's burden of software maintenance, and reduces the expense of software purchases.

Software-as-a-Service solutions are at the top end of the Cloud Computing stack and they provide end users with an integrated service comprising hardware, development platforms, and applications. Users are not allowed to customize the service but get access to a specific application hosted in the Cloud. Examples of SaaS implementations are the services provided by Google for office automation, such as Google Mail, Google Documents, and Google Calendar, which are delivered for free to the Internet users and charged for professional quality services. Examples of commercial solutions are SalesForce.com and Clarizen.com, which provide online CRM (Customer Relationship Management) and project management services, respectively [20].

3.2.4 Data-as-a-Service

Data in various formats and from multiple sources could be accessed via services by users on the network. Users could, for example, manipulate the remote data just like operate on a local disk or access the data in a semantic way in the Internet. Amazon Simple Storage Service (S3) provides a simple Web services interface that can be used to store and retrieve, declared by Amazon, any amount of data, at any time, from anywhere on the Web. The DaaS could also be found at some popular IT services, e.g., Google Docs and Adobe Buzzword. ElasticDrive is a distributed remote storage application which allows users to mount a remote storage resource such as Amazon S3 as a local storage device.

3.3 Cloud Vendors

Commercial Cloud Computing providers offer their services according to three fundamental service models. Table 2 summarizes how these three classes virtualize computation, storage, and networking [8].

Amazon EC2 is at one end of the spectrum. An EC2 instance looks much like physical hardware, and users can control nearly the entire software stack, from the kernel upwards. At the other extreme of the spectrum are application domain-specific platforms; such as Google AppEngine and Force.com, the SalesForce business software development platform. AppEngine is targeted exclusively at traditional web applications, enforcing an application structure of clean separation between a stateless computation tier and a stateful storage tier. Similarly, Force.com is designed to support business applications that run against the salesforce.com database, and nothing else. Microsoft's Azure is an intermediate point on this spectrum of flexibility vs. programmer convenience. Azure applications are written using the .NET libraries, and compiled to the Common Language Runtime, a language independent managed environment.

Aside the commercial Cloud vendors, another implementation of clouds is taking place, especially in the research context. Three main Clouds could be nominated as Open Source Clouds. The first Cloud Architecture, Eucalyptus, is an IaaS system with the aim of creating an open-source infrastructure architected specifically to support Cloud Computing research and infrastructure development. The system combines a Cloud Controller responsible for processing incoming user requests, manipulating the Cloud fabric and processing SLAs in company with a Client Interface that utilizes Internet standard protocols for instance HTTP XML and SOAP. Another Cloud architecture named Enomalism is another IaaS system that presents an organization with the capability to manage virtual infrastructures (including networks), virtual machine images and fine grained security and group management, in addition to the creation of virtual machine images. A third Cloud architecture, named OpenNebula, based on the research being performed by the

Table 2 Cloud computing vendors and how each providers virtualized resources

	Amazon Web Services	Microsoft Azure	Google AppEngine
Computation model (VM)	x86 Instruction Set Architecture (ISA) via Xen VM computation elasticity allows scalability, but developer must build the machinery, or third party VAR such as rightScale must provide it	Microsoft Common Language Runtime (CLR) VM; common intermediate form executed in managed environment machines are provisioned based on declarative descriptions (e.g. which roles can be replicated); automatic load balancing	Predefined application structure and framework; programmer-provided handlers written in Python, all persistent state stored in MegaStore (outside Python code) Automatic scaling up and down of computation and storage; network and server failover; all consistent with 3-tier Web app structure
Storage model	Range of models from block store (EBS) to augmented key/blob store (SimpleDB) Automatic scaling varies from no scaling or sharing (EBS) to fully automatic (SimpleDB, S3), depending on which model used Consistency guarantees vary widely depending on which model used APIs vary from standardized (EBS) to proprietary	SQL data services (restricted view of SQL Server) Azure storage service	MegaStore/BigTable
Networking model	Declarative specification of IPlevel topology; internal placement details concealed Security Groups enable restricting which nodes may communicate Availability zones provide abstraction of independent network failure Elastic IP addresses provide persistently routable network name	Automatic based on programmer's declarative descriptions of app components (roles)	Fixed topology to accommodate 3-tier Web app structure Scaling up and down is automatic and programmer invisible

Reservoir Project, the European research initiative in virtualized infrastructure and Cloud Computing, combines features of IaaS and PaaS in one architecture [19].

4 Cloud Computing Aspects

In this section, customer-related aspects; such as security, service pricing, Service Level Agreements and security; are presented.

4.1 Cost and Pricing

Cloud Computing providers have detailed costing models and metrics that are used to bill users on a pay-per-use basis. This makes it easy for users to see the exact costs of running their applications in the Cloud and it could well be that the design of their system can have a significant effect on its running costs.

From a Consumer Perspective, the three pricing models that are used by Cloud service providers are namely tiered pricing, per-unit pricing and subscription-based pricing. Tiered pricing is where different tiers each with different specifications (e.g. CPU and RAM) are provided at a cost per unit time. An example is Amazon EC2, where a small tier virtual machine has the equivalent of a 1.0 GHz CPU, 1.7 GB of RAM with 160 GB of storage and costs $0.085 per hour at the time of writing, whereas as a large tier machine has the equivalent of four 1.0 GHz CPUs, 7.5 GB of RAM with 850 GB of storage and costs $0.34 per hour 5. Per-unit pricing is where the user pays for their exact resource usage, for example it costs $0.15 per GB per month to store data on Amazon's Simple Storage Service (S3). Subscription-based pricing is common in SaaS products; such as Salesforce's Enterprise Edition CRM that costs $125 per user per month. Elasticity, which is the ability to quickly scale up or down one's resource usage, is an important economic benefit of Cloud Computing as it transfers the costs of resource over-provisioning and the risks of under-provisioning to Cloud providers. An often cited real-world example of elasticity is Animoto.com whose active users grew from 25,000 to 250,000 in three days after they launched their application on Facebook.

From a Provider Perspective, the cost is mainly of building Cloud data centers, which is different from building enterprise data centers. The costs of Cloud data centers can be reduced by looking at the cost of servers, infrastructure, power, and networking. For example, running data centers at hotter temperatures can reduce cooling costs and building micro data centers near users can also reduce bandwidth costs. A tool to model the Total Cost of Ownership (TCO) of setting up and maintaining a Cloud by taking into account the costs of hardware, software, power, cooling, staff and real-estate has been developed. The tool would probably be useful for both large and small Cloud providers as the cost factors are common to both. Calculating the TCO of a Cloud starts by taking the required number of physical servers as an input. Also, a method for calculating a cloud's Utilization Cost has been provided. This allows cloud providers to calculate costs in a similar manner to TCO. However, rather than inputting the number of physical servers into the model, they can start by inputting the maximum number of virtual machines that they need in their cloud. The model would then calculate the required number of physical servers and racks depending on the virtual machine density that is defined by the user (this is the maximum number of virtual machines that can be deployed on a physical server). Modeling the Utilization Cost can be useful for cloud providers as it allows them to see and analyze a detailed breakdown of the total costs of building a cloud [16].

Currently, providers follow a fairly simple pricing scheme to charge users fixed prices based on various resource types. For processing power (as of 15 October 2008), Amazon charges $0.10 per virtual computer instance per hour (h), Sun Microsystems charges $1.00 per processor (CPU) per h , and Tsunamic Technologies charges $0.77 per CPU per h. Charging fixed prices is simple to understand and straightforward for users, but does not differentiate pricing to exploit different user requirements in order to maximize revenue. Moreover, in utility computing it is not fair to both the provider and users since different users have distinctive needs and demand specific QoS for various resource requests that can change anytime. Instead of charging fixed prices for these heavy users, charging variable prices has been advocated providing guaranteed QoS through the use of advanced reservations. Advance reservations are bookings made in advance to secure an available item in the future and are used in the airline, car rental, and hotel industries. In the context of utility computing, an advance reservation is a guarantee of access to a computing resource at a particular time in the future for a particular duration. Hence, implementing autonomic metered pricing for a utility computing service to self-adjust prices to increase revenue becomes important. In particular, autonomic metered pricing can also be straightforward for users through the use of advanced reservations. With advanced reservations, users can not only know the prices of their required resources in the future ahead, but are also able to guarantee access to future resources to better plan and manage their operations [21].

4.2 Service Level Agreement

The Cloud constitutes a single point of access for all services which are available anywhere in the world, on the basis of commercial contracts that guarantee satisfaction of the QoS requirements of customers according to specific Service Level Agreements (SLAs). The SLA is a contract negotiated and agreed between a customer and a service provider. That is, the service provider is required to execute service requests from a customer within negotiated quality of service (QoS) requirements for a given price [1]. The purpose of using SLAs is to define a formal basis for performance and availability the provider guarantees to deliver. SLA contracts record the level of service, specified by several attributes; such as availability, serviceability, performance, operation, billing or even penalties in the case of violation of the SLA. Also, a number of performance-related metrics are frequently used by Internet Service Providers (ISPs); such as service response time, data transfer rate, round-trip time, packet loss ratio and delay variance. Often, providers and customers negotiate utility-based SLAs that determine the cost and penalties based on the achieved performance level. A resource allocation management scheme is usually employed with a view to maximizing overall profit (utility) which includes the revenues and penalties incurred when QoS guarantees are satisfied or violated, respectively. Usually, step-wise utility functions are used where the revenue depends on the QoS levels in a discrete fashion [22].

The service providers and their customers negotiate utility based SLAs to determine costs and penalties based on the achieved performance level. The service provider needs to manage its resources to maximize its profits. Recently, the problem of maximization of SLA revenues in shared service center environments implementing autonomic self-managing techniques has attracted vast attention by the research community. The SLA maximization problem considers (1) the set of servers to be turned ON depending on the system load, (2) the application tiers to servers' assignment, (3) the request volumes at various servers, and (4) the scheduling policy at each server as joint control variables [23].

4.3 Security

Certainly as the Cloud Computing environment matures, security concerns are being addressed more consistently. In October 2008, Google unveiled a new service-level agreement (SLA) that guarantees 99.9 % system accessibility for users of its Google Apps Premier Edition, joining the Amazon S3 SLA that launched in 2007. Other companies are addressing related angles of security. For example, Boulder, Colo.-based Symplified offers a unified access management system built for the Cloud architectures of SaaS. The company's Identity-as-a-Service approach provides on-demand features; such as single sign-on, provisioning, de-provisioning, auditing and compliance, access and authorization, identity synchronization, strong authentication, and identity integration [7].

One of the most serious concerns is the possibility of confidentiality violations. Either maliciously or accidentally, Cloud provider's employees can tamper with or leak a company's data. Such actions can severely damage the reputation or finances of a company. In order to prevent confidentiality violations, Cloud services' customers might resort to encryption. While encryption is effective in securing data before it is stored at the provider, it cannot be applied in services where data is to be computed, since the unencrypted data must reside in the memory of the host running the computation [14].

The Cloud also offers security advantages. For example, intruders have no access to the source code, and providers often work hard to provide clean, unbreakable barriers between customers. Security can differ greatly from application to application, from platform to platform, and from provider to provider, however. Yet on the whole, the Cloud holds much promise for better security.

Beyond generic concerns regarding the Cloud approach, each of the three models has its own security concerns.

With SaaS, users must rely heavily on their Cloud providers for security. The provider must do the work to keep multiple companies or users from seeing each other's data without permission. In addition, the provider must protect the underlying infrastructure from break-ins and generally has responsibility for all authentication and encryption. It's difficult for the customer to get the details that help ensure that the right things are being done. Similarly, it's difficult to get assurance that the application will be available when needed.

With PaaS, the provider might give some control to the people building applications atop its platform. For example, developers might be able to craft their own authentication systems and data encryption, but any security below the application level will still be completely in the provider's hands. Usually, the provider will offer little or no visibility into its practices. Plus, the platform provider must be able to offer strong assurances that the data remains inaccessible between applications. Large banks don't want to run a program delivered in the Cloud those risks compromising their data through interaction with some other program.

With IaaS, the developer has much better control over the security environment, primarily because applications run on virtual machines separated from other virtual machines running on the same physical machine, as long as there is no gaping security hole in the virtual machine manager. This control makes it easier to ensure that developers properly address security and compliance concerns, with the downside that building the application can be more expensive and time-consuming.

Backing up data poses another concern. Even though some providers do their own backups for the customer, much can still go wrong. Maybe they increase their prices and make it difficult to get data off their network [24].

5 Cloud Computing and Distributed Computing Paradigms

The emergence of Cloud Computing continues the natural evolution of Distributed Systems to cater for changes in application domains and system requirements [19].

Cloud Computing builds on top of several other technologies, i.e. Distributed Computing, Grid Computing, Utility Computing and Autonomic Computing, and it can be envisaged as a natural step forward from the Grid-utility model. In the heart of Cloud Computing infrastructure, a group of reliable services are delivered through powerful data computing centers that are based on modern virtualization technologies and related concepts such as component-based system engineering, orchestration of different services through workflows and Service-Oriented Architectures (SOAs) [22].

Cloud Computing is based on the use of distributed computing resources that are easily allocated, de-allocated, migrated and possibly re-allocated on user request. As such, it relies heavily on the use of virtualization technologies, able to offer an almost unlimited amount of virtual computing resources. Virtualization controls the access to physical resources in a transparent way, it is possible to offer computational resources with full control, in that final users can configure them as administrators, without any restriction [25].

5.1 Utility Computing Versus Cloud Computing

Cloud services should be accessed with simple and pervasive methods. In fact, the Cloud Computing adopts the concept of Utility Computing. In other words, users obtain and employ computing platforms in computing Clouds as easily as they access a traditional public utility (such as electricity, water, natural gas, or telephone network) [5]. In detail, the Cloud services enjoy the following features:

- The Cloud interfaces do not force users to change their working habits and environments, e.g., programming language, compiler and operating system. This feature differentiates Cloud Computing from Grid computing as Grid users have to learn new Grid commands and APIs to access Grid resources and services.
- The Cloud client software which is required to be installed locally is lightweight. For example, the Nimbus Cloudkit client size is around 15 MB.
- Cloud interfaces are location independent and can be accessed by some well-established interfaces like; Web services framework and Internet browser.

5.2 Volunteer Computing Versus Cloud Computing

Both Cloud and volunteer computing have similar principles, such as transparency. On both platforms, one submits tasks without needing to know the exact resource on which it will execute. However, in practice, the Cloud Computing infrastructures differ from volunteer computing platforms throughout the hardware and software stack. From the perspective of the user, there are two main differences, namely configurability (and thus homogeneity), and quality-of-service. Clouds present a configurable environment in terms of the OS and software stack with the Xen virtual machine forming the basis of EC2. So, while Clouds can offer a homogeneous resource pool, the heterogeneity of VC hardware and operating system is not transparent to VC application developers.

Clouds also provide higher quality-of-service than VC systems. Cloud resources appear dedicated, and there is no risk of preemption. Many Cloud Computing platforms, such as Amazon's EC2, report several nine's in terms of reliability. Cloud infrastructures consist of large-scale centralized compute servers with network-attached storage at several international locations. The infrastructures are accessed through services such as S3 also provide high-level web services for data management [26].

5.3 Grid Computing Versus Cloud Computing

Cloud Computing, born in the e-business context, and Grid computing, originated in the e-science context, are two different but similar paradigms for managing large sets of distributed computing resources [25].

Grid computing can be seen as one of several predecessors to Cloud Computing. Grid computing is often about making large computations using large amounts of resources, while as Cloud Computing is more about making large amounts of resources available to many different applications over a longer period of time. Clouds leverage modern technologies such as virtualization to provide the infrastructure needed to deploy services as utilities. Still, Cloud Computing and Grid computing share a lot of the underlying technology and many concepts from Grid computing can be modified and made suitable for Cloud Computing as well [27].

A comparison between Grid and Cloud is shown hereunder in the following aspects [5]:

- *Definition*: Grid computing, oriented from high performance distributed computing, aims to share distributed computing resource for remote job execution and for large scale problem solving. The Grid computing emphasizes the resource side by making huge efforts to build an independent and complete distributed system. Cloud Computing provides user-centric functionalities and services for users to build customized computing environments. Cloud Computing, which is oriented towards the industry, follows an application-driven model.
- *Infrastructure*: Grid infrastructure enjoys the following features. Grid infrastructure in nature is a decentralized system, which spans across geographically distributed sites and lack central control. Grid infrastructure normally contains heterogeneous resources, such as hardware/software configurations, access interfaces and management policies. On the contrary, from the viewpoint of users, computing Clouds operate like a central compute server with single access point. Cloud infrastructures could span several computing centers, like Google and Amazon, and in general contain homogeneous resources, operated under central control.

- *Middleware*: Grid computing has brought up full-fledged middleware, for example, Unicore, Globus Toolkit and gLite. The Grid middleware could offer basic functionalities like resource management, security control and monitoring & discovery. The Grid community has established well-defined industry standards for Grid middleware, e.g.,WSRF. The middleware for Cloud Computing, or the Cloud operating system, is still underdeveloped and lacks of standards. A number of research issues remain unsolved, for example, distributed virtual machine management, Cloud service orchestration, and distributed storage management.
- *Accessibility and application*: Grid computing has the ambitious objective to offer dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities. However inexperienced users still find difficulties to adapt their applications to Grid computing. Furthermore, it is not easy to get a performance guarantee from computational Grids. Cloud Computing, on the contrary, could offer customized, scalable and QoS guaranteed computing environments for users with an easy and pervasive access. Grid computing has gained numerous successful stories in many application fields. A recent famous example is LCG project to process huge data generated from the Large Hadron Collider (LHC) at CERN. Although Amazon EC2 has obtained a beautiful success, more killer applications are still needed to justify the Cloud Computing paradigm.

Based on the fore mentioned discussion, it could be concluded that Grid computing has established well organized infrastructures, middleware and application experience. Cloud Computing recently outpaces Grid computing because it could offer computing environments with desirable features, such as QoS guarantee and computing environment customization. It would be a good solution to build Cloud Computing services on Grid infrastructure and middleware, thus providing users with satisfying services.

Table 3 compares different features of Grid and Cloud highlighting the similarities and differences between both paradigms.

In fact, Cloud and Grid computing paradigms have many points in common: both adopt large datacenters, both offer resources to users, both aim at offering a common environment for distributed resources. At the state of the art, there are two main approaches for their integration:

- *Grid on Cloud*: a Cloud approach (i.e. adoption of virtual machines) is adopted in order to build up and to manage a flexible Grid system. As in this context the Grid middleware runs on a virtual machine, the main drawback of this approach is performance. Virtualization inevitably entails performance losses as compared to the direct use of physical resources.
- *Cloud on Grid*: the complex and stable Grid infrastructure is exploited to build up a Cloud environment. This is a more common solution. In this case, a set of Grid services is offered in order to manage virtual machines. The use of Globus workspaces, a set of Grid services for the Globus Toolkit 4, is the prominent

Table 3 Grid versus gloud characterstics

Feature	Grid	Cloud
Resource Sharing	Collaboration (VOs, fair share)	Assigned resources are not shared
Resource heterogeneity	Aggregation of heterogeneous resources	Aggregation of heterogeneous resources
Virtualization	Virtualization of data and computing resources	Virtualization of hardware and software platforms
Security	Security through credential delegations	Security through isolation
High level services	Plenty of high level services	No high level services defined yet
Architecture	Service orientated	User chosen architecture
Software dependencies	Application domain-dependent software	Application domain-independent software
Platform awareness	The client software must be grid-enabled	The SP software works on a customized environment
Software workflow	Applications require a predefined workflow of services	Workflow is not essential for most applications
Scalability	Nodes and sites scalability	Nodes, sites, and hardware scalability
Self-management	Re-configurability	Re-configurability, self-healing
Centralization degree	Decentralized control	Centralized control (until now)
Usability	Hard to manage	User friendliness
Standardization	Standardization and interoperability	Lack of standards for Clouds interoperability
User access	Access transparency for the end user	Access transparency for the end user
Payment Model	Rigid	Flexible
QoS Guarantees	Limited support, often best-effort only	Limited support, focused on availability and uptime

solution. It is also adopted in the Nimbus project, where a Cloud environment is built on top of a Grid.

Both Grid Computing and Cloud Computing implement the more general model of Utility Computing. However, the motivation and approaches to realize an infrastructure that serves compute utilities are considerably different. Grid Computing evolved in a scientific community as a means to solve computationally challenging tasks in science and industry. Although Clouds can also be used for large-scale batch jobs, they focus on very different applications. Emerging in a commercial environment, the main purpose of Cloud Computing is to provide a platform to develop, test and deploy Web-scale applications and services [9]. The motivation behind research into Grid Computing was initially the need to manage large scale resource intensive scientific applications across multiple administrative domains that require many more resources than that can be provided by a single computer. Cloud Computing shares this motivation, but within a new context oriented towards business rather than academic resource management, for the stipulation of reliable services rather than batch oriented scientific applications [19].

6 The Road Ahead

Despite the fact that Cloud Computing promises to bring sweeping changes to the way of using information technology, decisions-makers are still struggling for insights on the challenges associated with trying to exploit it.

In this section, we present the lessons learned so far, discuss the challenges that should be addressed in future research directions.

6.1 Lessons Learned in Cloud Adoption

No two organizations had adopted Cloud Computing in exactly the same way or over the same timeframe. So far, the journeys followed were determined by history and accidents (experiments, unplanned trials, recommendations from peers) as well as variations in the structure, industrial context and computing needs of individual organizations. Two main themes, however, did emerge as important, being broadly applicable to multiple cases. In one of these the journey was initiated by the adoption of a Cloud-based service for a single key enterprise application. Good results then led to the adoption of more Cloud-based software development to extend the functionality of standard applications. In the other, the journey began with efforts to improve the capacity utilization of computing resources inside the organization. Servers and computer storage were considered around fewer, larger facilities, and some software applications were virtualized, so they ran anywhere in these facilities rather than on dedicated machines. Good results saw this expanded into an organization-wide strategy, followed by a shift to service providers outside the organization.

There are also many other variations; in one case the journey began with adoption of open source software (installed on premises). The associated shift from up-front software purchasing to pay-as-you-go maintenance led to a broader interest in paying for computer resources using utility-based pricing. In another case, the adoption of web-based email and collaboration tools started the process. In third case, the journey began with a desire to remove computing equipment from hostile work environments. As with the journeys taken, there was a wide variation in the extent of adoption.

6.2 Challenges and Future Research Directions

With the flourish of the Cloud Computing and the wide availability of both providers and tools, a significant number of challenges are facing such new paradigm. These challenges are the key points to be considered for future research directions. Challenges to be faced include user privacy, data security, data lock-in, availability, disaster recovery, performance, scalability, energy-efficiency, and programmability. Some of these challenges are further described as follows:

- *Security, Privacy, and Trust*: With the movement of the clients' data into the Cloud, providers may choose to locate them anywhere on the planet. The physical location of data centers determines the set of laws that can be applied to the management of data [4]. Security and privacy affect the entire Cloud Computing stack, since there is a massive use of third-party services and infrastructures that are used to host important data and to perform critical operations. In such scenarios, trusting the provider is a fundamental issue to ensure the desired level of privacy [28]. Accordingly, it becomes crucial to give deep attention to such legal and regulatory perspectives.
- *Data Lock-Inand Standardization*: A major concern of Cloud Computing users is about having their data locked-in by a certain provider. Users may want to move data and applications out from a provider that does not meet their requirements. However, in their current form, Cloud Computing infrastructures and platforms do not employ standard methods of storing users' data and applications. Hence, the interoperate option is not yet supported and users' data are not portable. This is why a lot of work is requested in terms of standardization and supporting interoperation option.
- *Availability*, *Fault-Tolerance*, *and Disaster Recovery*: It is expected that users will have certain expectations about the service level to be provided once their

applications are moved to the Cloud. These expectations include availability of the service, its overall performance, and what measures are to be taken when something goes wrong in the system or its components. Generally, users seek for a warranty before they can comfortably move their business to the cloud that is Service Level Agreement (SLA). SLAs, which include QoS requirements, must be ideally settled to cover many issues such as the service availability, the fault-tolerance, and the disaster recovery. This is why this topic is still hot and in need for a lot of effort to be professionally covered.

- *Resource Managementand Energy-Efficiency*: One important challenge faced by providers of Cloud Computing services is the efficient management of virtualized resource pools. Physical resources such as CPU cores, disk space, and network bandwidth must be sliced and shared among virtual machines running potentially heterogeneous workloads [29]. In addition to optimize application performance, dynamic resource management can also improve utilization and consequently minimize energy consumption in data centers. This can be done by judiciously consolidating workload onto smaller number of servers and turning off idle resources. Researches in this area are endless and still in need for more.

7 Conclusion

This chapter presents an overview on the Cloud Computing paradigm. First, the Cloud Computing anatomy has been described then it has been defined from different perspectives; such as the service and business models as well as its economical and quality of service aspects. The Cloud Computing distinguishes itself from other computing paradigms, by its promising features including on-demand, flexible and QoS guaranteed provisioned service. The affects of Cloud Computing on organizations and enterprises were also addressed in terms of time, management and operational costs. The architecture of Clouds has been discussed; differentiating between the Cloud core stack and the management. The first has three layers in the core stack: resource, platform and application layers; while the later includes managerial tasks such as billing and scheduling. Enabling technologies contributing to Cloud Computing; such as virtualization and web 2.0, web services and SOA, were also clarified.

Second, the Cloud Computing classification is addressed, clearly differentiating between deployment and service models apart, and beneath each of them. That is Clouds are classified according to their deployment models as private, community, public and hybrid Clouds. Clouds are also classified according to their offered service models; infrastructure, platform and software as services. Examples of the Cloud providers and vendors are listed such as Amazon Web Services, Microsoft Azure and Google AppEngine provide the above mentioned services respectively.

Next, the customer-related aspects were discussed including the pricing and costing models with metrics that are used to bill users on a pay-per-use basis. The Service Level Agreement, which is a contract between the customer and the service provider, was also addressed as well as the data security perspectives.

Furthermore, a comprehensive comparison between Cloud Computing and utility, volunteer and grid paradigms has been conducted; in terms of architecture, resources sharing, QoS guarantees and security. The analysis of the Cloud reports that such technology is based on the use of distributed computing resources that are easily allocated, de-allocated, migrated and possibly re-allocated based on the user request.

Finally, to guarantee the long-term success of Cloud Computing, the chapter tackles some significant challenges that face the Cloud paradigm. Challenges that need to be carefully addressed for future research like; user privacy, data security, data lock-in, availability, disaster recovery, performance, scalability, energy-efficiency, and programmability.

By that, this chapter has provided a comprehensive overview on the newly emerged Cloud Computing paradigm and paved the way for further research in this area.

References

1. K. Xiong, H. Perros, Service performance and analysis in cloud computing, in *Proceedings of Congress on Services - I (SERVICES 09)*. IEEE Computer Society Washington 2009, 693–700
2. Y. Sun, J. White, J. Gray, A. Gokhale, D. Schmidt, Model-Driven Automated Error Recovery in Cloud Computing, in *Model-driven Analysis and Software Development: Architectures and Functions*, ed. by J. Osis, E. Asnina (IGI Global, Hershey, PA, USA, 2011), pp. 136–155
3. R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud Computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Gener. Comput. Syst. **25**, 599–616 (2009)
4. W. Voorsluys, J. Broberg, and R. Buyya, Introduction to Cloud Computing, in Cloud Computing: Principles and Paradigms, ed. by R. Buyya, J. Broberg, A.Goscinski (New York: Wiley, 2011), pp. 1–41
5. L. Wang, G. Laszewski, Scientific cloud computing: Early definition and experience, in Proceedings of 10th IEEE International Conference on High Performance Computing and Communications (Dalian, China, 2008), pp. 825–830
6. J. Geelan, Twenty one experts define cloud computing. Virtualization Mag. (2010)
7. N. Kho, Content in the cloud. EContent Mag. (2009)
8. M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, Above the Clouds: A Berkeley View of Cloud Computing, University of California at Berkley, USA, Technical Rep UCB/EECS-2009-28, (2009)
9. M. Klems, J. Nimis, and S. Tai, Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing", in Designing E-Business Systems: Markets, Services, and Networks, vol. 22, C. Weinhardt, S. Luckner, and J. Stober, Eds. Heidelberg: (Springer Berlin, 2009), part 4, pp. 110–123

10. A. Tikotekar, G. Vallee, T. Naughton, H. Ong, C. Engelmann, S. Scott, A. Filippi, *Effects of virtualization on a scientific application running a hyperspectral radiative transfer code on virtual machines, in Proceedings of 2nd Workshop on System-Level Virtualization for High Performance Computing (HPCVirt 08)* (Glasgow, UK, 2008), pp. 16–23
11. R. Buyya, C. Yeo, S. Venugopal, *Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities, in Proceedings of 10th IEEE International Conference on High Performance Computing and Communications (HPCC 08)* (Dalian, China, 2008), pp. 5–13
12. P. McFedries, The cloud is the computer. IEEE Spectrum **45**(8), 20–20 (2008)
13. P. Mell, T. Grance, *The NIST Definition of Cloud Computing* (National Institute of Standards and Technology, Information technology laboratory, 2009)
14. N. Santos, K. Gummadi, R. Rodrigues, *Towards trusted cloud computing, in Proceedings of conference on Hot topics in Cloud Computing (HOTCLOUD 09)* (San Diego, USA, 2009)
15. J. Grimes, P. Jaeger, J. Lin, *Weathering the storm: The policy implications of cloud computing, in Proceedings of iConference, University of North Carolina* (Chapel Hill, USA, 2009)
16. A. Khajeh-Hosseini, I. Sommerville, and I. Sriram, Research challenges for enterprise cloud computing, CoRR, abs/1001.3257, 2010
17. L. Qian, Z. Luo, Y. Du, L. Guo, *Cloud computing: An overview, in Proceedings of 1st International Conference on Cloud Computing* (Beijing, China, 2009), pp. 626–631
18. F. Aymerich, G. Fenu, S. Surcis, *A real time financial system based on grid and cloud computing, in Proceedings of ACM symposium on Applied Computing* (Honolulu, Hawaii, USA, 2009), pp. 1219–1220
19. R. Buyya, R. Ranjan, R. Calheiros, *InterCloud: Utility-oriented federation of cloud computing environments for scaling of application services, in Proceedings of 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP* (Busan, Korea, 2010), pp. 13–31
20. R. Buyya, S. Pandey, C. Vecchiola, *Cloudbus toolkit for market-oriented cloud computing, in Proceedings of 1st International Conference on Cloud Computing (CLOUDCOM 09)* (Beijing, China, 2009), pp. 24–44
21. C. Yeoa, S. Venugopalb, X. Chua, R. Buyya, Autonomic metered pricing for a utility computing service. J. Future Gener. Comput. Syst. **26**(8), 1368–1380 (2010)
22. C. Yfoulis, A. Gounaris, *Honoring SLAs on cloud computing services: a control perspective, in Proceedings of European Control Conference* (Budapest, Hungary, 2009), pp. 184–189
23. D. Ardagna, M. Trubianb, L. Zhangc, SLA based resource allocation policies in autonomic environments. J. Parallel Distrib. Comput. **67**(3), 259–270 (2007)
24. J. Viega, Cloud Comput. Common Man. J. Comput. 42(8), 106–108 (2009). IEEE Computer Society
25. E. Mancini, M. Rak, U. Villano, PerfCloud: GRID services for performance-oriented development of cloud computing applications", in Proceedings 18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE 09). (Groningen, The Netherlands, 2009), 201–206 (2009)
26. D. Kondo, B. Javadi, P. Malecot, F. Cappello, amd D. Anderson, Cost-benefit analysis of cloud computing versus desktop grids, in Proceedings of IEEE International Symposium on Parallel and Distributed Processing (IPDPS 09), IEEE Computer Society Washington, 2009, pp. 1–12
27. E. Elmroth, F. M'arquezy, D. Henriksson, D. Ferrera, *Accounting and billing for federated cloud infrastructures, in Proceedings of Eighth International Conference on Grid and Cooperative Computing (GCC 09)* (Lanzhou, China, 2009), pp. 268–275
28. R. Buyya, S. Pandey, C. Vecchiola, Cloudbus toolkit for market-oriented cloud computing, in Proceedings of 1st International Conference on Cloud Computing (CLOUDCOM 09), (Beijing, 2009), pp. 24–44

29. W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, Cost of virtual machine live migration in clouds: A performance evaluation, in Proceedings of 1st International Conference on Cloud Computing, (Beijing, 2009), pp. 254–265
30. L. Vaquero, L. Rodero-Merino, J. Caceres, M. Lindner, A break in the clouds: Towards a cloud definition. ACM SIGCOMM Comput. Commun. Rev. **9**(1), 50–55 (2009)
31. D. Armstrong, and K. Djemame, Towards quality of service in the cloud, in Proceedings of 25th UK Performance Engineering Workshop, (Leeds, 2009)

Springer
http://www.springer.com/978-3-642-35015-3

Inter-cooperative Collective Intelligence: Techniques
and Applications
Xhafa, F.; Bessis, N. (Eds.)
2014, XV, 440 p., Hardcover
ISBN: 978-3-642-35015-3