CLOUD SCALABILITY CONSIDERATIONS

Maram Mohammed Falatah[1], Omar Abdullah Batarfi[2]

Department of Computer Science, King Abdul Aziz University, Saudi Arabia

ABSTRACT

Cloud computing is a technique that has a great capabilities and benefits for users. Cloud characteristics encourage many organizations to move to this technology. But many consideration faces transmission process. This paper outline some of these considerations and considerable efforts solved cloud scalability issues.

KEYWORDS

Cloud Computing, Scalability, Grid ,Virtualization

1. Introduction

Cloud Computing resources and computing power are made available through distributed and sharing services virtually. Through Cloud Computing, services can be updated to cope with the rate at which the volume of data on the Internet is growing. Virtualization technique [1] integrates resources from a huge computation and storage network, such that users only need one low-cost device for accessing the network. Users can access resources and services without having to consider their sources a typical situation for Internet services. However, moving to such technique require awareness about performance considerations which are then described.

2. Service Level

As mentioned, users must address concerns about moving to a new technology. When an organization [2] requires a service—whether from the cloud or from a traditional data centre—it generally drafts a service-level agreement that identifies key metrics, called service levels, that the organization can reasonably expect from the service. The ability to understand and to fully trust the availability, scalability and performance of the cloud is key for many technologists interested in moving into the cloud. The discussion in the following sections will focus in particular on those aspects related to scalability issues.

3. Cloud Scalability Issues

As cloud advantage, Cloud Computing is a scalable and easy way for users to access a large pool of virtualized resources that can be dynamically provisioned to adjust to a variable workload. But first, it is useful to define scalability term and illustrate cloud scalability among three cloud services.

‘Scalability’ can be defined in different ways. It can define as [3] "the ability of a particular system to fit a problem as the scope of that problem increases (number of elements or objects, growing volumes of work and/or being susceptible to enlargement)." Also can defined as [2] "Scalability of service is a desirable property of a service which provides an ability to handle growing amounts of service loads without suffering significant degradation in relevant quality

attributes. The scalability enhanced by scalability assuring schemes such as adding various resources should be proportional to the cost to apply the schemes." Another definition state that [4] "Scalability is the ability of an application to be scaled up to meet demand through replication and distribution of requests across a pool or farm of servers."

Previous definitions conclude that scalability is about holding unexpected workloads, and it depends on system design, as well as the types of data structures, algorithms and communication mechanisms used to implement system components.

In addition, scalability should be transparent to users without involving them in any details. For example, users should be able to store their data in the cloud without needing to know where it is kept or how they are accessing it. This scalability can be performed in the cloud through different levels.

4. Scalability Levels

Scalability is one of the major advantages of the cloud paradigm. More specifically, it is the advantage that distinguishes clouds from advanced outsourcing solutions. However, some important pending issues must be addressed before the dream of automated scaling of applications can be realized. The most notable initiatives towards whole application scalability in cloud environments are as follow [5]:

4.1. Server Scalability

Most available Infrastructure as a Service (IaaS) clouds work with individual Virtual Machine (VM) management primitives—such as elements for adding or removing VMs—but lack mechanisms for treating applications as single entities or for managing relationships among application components. For example, relationships between VMs are often not considered, ordered deployment of VMs containing software for different tiers of an application is not automated (e.g. the database IP is only known at deployment time, so the database must be deployed first in order to get its IP and configure the web server that connects to it). Application providers typically manage only applications, not virtual infrastructure terms.

4.2. Scaling of the Network

Networking over virtualized resources is typically done in two different ways: Ethernet virtualization and overlay networks and TCP/IP virtualization. Separation of user traffic is not enough for complete application scalability: the need to scale the network arises in consolidated data centers that host several VMs per physical machine. Scalability is often achieved by over-provisioning resources to meet this increased demand.

4.3. Scaling of the Platform

IaaS clouds give application providers a convenient way to control the resources used by their systems. However, IaaS clouds require that the application developers or system administrators install and configure the entire software stack that the application components need. In contrast, Platform as a Service (PaaS) clouds offer ready-to-use execution environments and convenient services for applications. Therefore, when using PaaS clouds, developers can focus on programming their components rather than on setting up the environments that the components require. However, because PaaS clouds may experience high usage PaaS providers must be able to scale execution environments accordingly.

5. Performance and Scalability Consideration

Cloud applications [5] should be able to request not only virtual servers at multiple points in the network, but also network pipes for provisioning bandwidth and other network resources to interconnect them in network as a service (NaaS). Clouds that offer simple virtual hardware infrastructure such as VMs and networks are, as has been mentioned, usually referred to as IaaS clouds. To get best quality from cloud performance, applications have to classify and designed according to following information:

5.1. Application Characteristics

Migration from a local network to external resources—such as a cloud—according to the actual demand is a major issue for companies, because the features of their applications differ. Many techniques are used to resolve this difficulty.

Some approaches can be used to choose which applications should be migrated [6]. These include focusing on particular applications, developing a profile for commonly used applications and choosing the top *N* applications. Distinguishing between critical applications and normal ones is important, because critical applications have the highest value in terms of performance requirements.

Likewise, understanding when peak data flow occurs may help in focus effort and resources during this period. As an example, if a company experiences peak flow before holidays, then it needs the maximum capability from its resources at those times. Calculation of peak periods is also the most important factor in identifying the worst case scenario and a typical usage scenario.

5.2. Designing Applications

Organizations should monitor some properties of applications. In this way, they can avoid problems when running the applications in the cloud. These properties are the IDEAL properties: the Isolation state, Distribution, Elasticity, Automated management and Loose coupling [7]. Cloud-native applications can be defined using these properties:

- **Isolated state**: a concept that is closely related to elasticity is the designing of large portions of a cloud application to be stateless; in this way, state is isolated in small portions of the application. Cloud providers, therefore, often restrict where application state may be handled in automatically scaled applications.
- **Distribution:** by nature, cloud environments are large, possibly globally-distributed environments that consist of many IT resources. Therefore, cloud applications must be decomposed into separate application components that can be distributed among resources in the environment.
- **Elasticity:** Cloud applications should by scaled out instead of scaled up. In this way increasing workload can be addressed by increasing the number of resources assigned to a customer or an application, not the capabilities of individual resources.
- **Automated management:** Due to the elasticity of cloud applications, resources are constantly added and removed during runtime. These tasks should be automated by monitoring system load and interacting with management interfaces of cloud providers to provision or decommission resources.
- **Loose coupling:** Because the number of IT resources on which a cloud application relies changes constantly, the dependencies between application components should be minimized. This reduces the need for provisioning and decommissioning tasks and also reduces the impact of the failure of application components.

5.3. IaaS and Applications

The underlying infrastructure and environment of a cloud must be designed and implemented in such a way that it is flexible and scalable [8]. Unfortunately, the history of the design, delivery and management of very-large-scale federally-developed systems does not offer many success stories to build upon. If a company does not implement the system properly, it risks significant challenges and costs in migrating information to different technologies as the third-party vendor upgrades its processing and storage environment. If this type of upgrade is managed in-house, resident IT professionals can more readily manage migration and harmonize data, users and processes. But the procedures that a cloud vendor executes in scaling its environment are managed without the input of customers, and may change services that the customer requires. Customers require the ability to increase bandwidth, speed and response time. In some cases, the cost of moving data to a cloud infrastructure has proven quite high in terms of time (bandwidth) and money. Some cloud users have resorted to using physical media to send data in order to expedite changes in their business needs.

Therefore, before a company constructs information infrastructure based on IaaS, it should consider some factors, such as physical environment, storage virtualization requirements, cost performance requirements and employee technical ability [9]. The main steps for implementing IaaS are as follows.

1. Collect the parameters of the infrastructure before IaaS. This is the basis for all subsequent work. All application services and host servers, operating systems and system resources must be collected.
2. Select servers for virtualization and estimate their requirements. Not all applications are suitable for running on a virtual machine. For example, video services are not suitable. After determining the servers to be virtualized, you should immediately assess their requirements in terms of CPU, memory, storage, networking, etc.
3. Select the appropriate server virtualization software platform and hardware facilities. This is the most critical step in the implementation of IaaS.
4. Create a construction plan. Campus network situations vary widely, and thus so does demand for the implementation of IaaS. Consider all possible conditions before implementing IaaS.
5. Buy software and hardware facilities. Most colleges and universities purchase equipment through government procurement channels. Because the government procurement cycle is generally long, it is best to buy software and hardware while the plan mentioned in step 4 is being created.
6. Deploy the necessary hardware and software platform. Hardware facilities should be deployed first, and then the software platform.
7. Virtualized the servers selected in step 2 according to the plan made in step 4. Actions to take include migration, system configuration, virtual-server backup and so on.
8. Assess and optimize the implementation. After a certain period of time following the implementation, the result should be assessed. Then, an optimization plan based on the results and the original implementation should be created and followed.

5.4. Proactive and Reactive Scaling

Proactive scaling is usually done in a cloud by scaling at predictable, fixed intervals or when big surges of traffic requests are expected [5][10]. A well-designed proactive scaling system enables providers to schedule capacity changes that match the expected changes in application demand. To perform proactive scaling, they should first understand expected traffic flow. This simply means that they should understand (roughly) how much normal traffic deviates from agency expectations. The most efficient use of resources is just below maximum agency capacity, but

scheduling things that way can create problems when expectations are wrong—even when reactive scaling is also implemented.

When a company begins running applications in cloud resources, some unexpected changes in the workload may occur. A reactive scaling strategy [10] can meet this demand by adding or removing scaling up or down resources. Periodic acquisition of performance data is important both to the cloud provider and to the cloud agencies for maintaining QoS.

In addition, reactive scaling enables a provider to react quickly to unexpected demand. The crudest form of reactive scaling is utilization-based. In other words, when CPU or RAM or another resource reaches a certain level of utilization, the provider adds more of that resource to the environment.

6. Application Scalability Researches

Automatic scalability at the application level can be implemented in several ways. The following paragraphs describe significant research, starting from grid and web services and continuing through the appearance of Cloud Computing.

6.1. Assuring High Scalability in Cloud Computing

Lee & Kim presented software-oriented approaches for ensuring the high scalability of services in Cloud Computing, [2] High scalability under high service loads does not come free of charge; it can be ensured by adopting some scalability assurance schemes. The most common conventional scheme is to simply add the required resources. But as they propose in the paper, other schemes can assure high scalability.

There is always some cost involved in running scalability assuring schemes, such as the cost of additional CPU and memory. The scalability gained through such schemes should be proportional to the cost of applying the schemes. That is, cost-effectiveness should be considered in assuring scalability. Services should provide the level of quality specified in their SLAs. Services with acceptable scalability should not suffer from significant degradation of QoS. Scalability assurance schemes should ensure that services satisfy the constraints of meeting the minimal threshold of their QoS attributes. From this, two effective software-oriented schemes can be derived: service replication and service migration.

- **Service replication:** Service replication is a technique for cloning services that are already running on the other nodes to optimize the service load over the nodes without affecting operations in progress. Replicated services secure additional resources provided by the new nodes for handling larger service load. In other words, service replication enhances service scalability and reduces the risk of QoS degradation by handling larger service loads. To perform a case study, they firstly set a service load as variable. The service load is a number of service invocations within a unit time. For the case study, we set 500ms as the unit time. That is, if ten invocations occur within 500ms, then the service load is 10. To show an effectiveness of service replication, they simulate the service replication scheme for the seventeen different volumes of service load. On each service load, they compare (1) conventional service system with (2) service replication scheme in terms of average response time. Table 1 shows the result of service replication test.

Table 1. result of service replication

ID	Svc Load	(1)	(2)
RE-01	10	11.96	12.96
RE-02	100	16.95	17.95
RE-03	200	29.148	28.268
RE-04	300	35.4847	35.4546
RE-05	400	36.2275	35.3452
RE-06	500	40.47	39.291
RE-07	600	41.2457	39.321
RE-08	700	41.5297	40.129
RE-09	800	41.9813	40.13
RE-10	900	43.0039	40.234
RE-11	1000	45.9874	40.432
RE-12	1100	45.7055	40.557
RE-13	1200	47.8391	40.912
RE-14	1300	46.6424	41.001
RE-15	1400	48.6932	41.124
RE-16	1500	49.7995	41.145
RE-17	2000	51.3759	43.5412
RE-18	3000	52.4973	45.3618

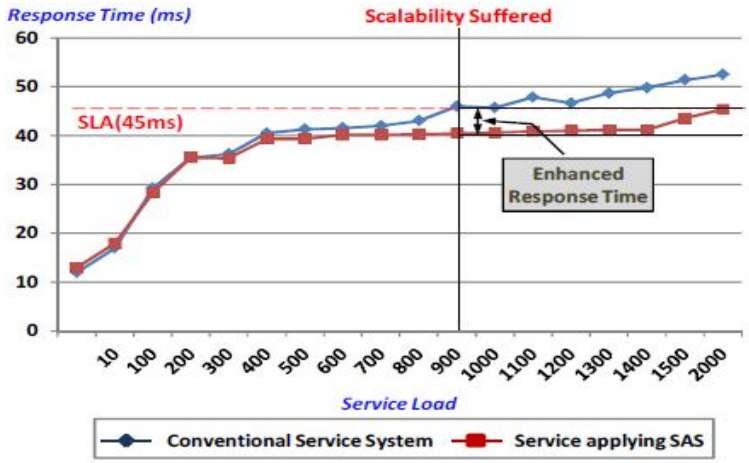

Figure 1. Chart of service replication result

- **Service migration:** Service migration is a scheme for placing a service on an alternative node when a particular node cannot provide high QoS due to a physical problem or software problem. After service migration, the migrated service performs the same role as it did on the unstable node, and the unstable node is removed from the list of service nodes. Table 2 shows the result for service migration. Variables, service load and appliance of scalability assuring scheme, are same with service replication simulation. In this simulation, they have a scenario that a service is migrated to a closer node from consumers. To enable this, they assume that the response time is proportional to the

distance. Therefore, a service is migrated to the fastest node in terms of the response time.

Table 2. Result of service migration

ID	Svc Load	(1)	(2)
MI-01	10	22.4	12.29
MI-02	100	26.154	14.75
MI-03	200	38.1	27.452
MI-04	300	39.665	34.154
MI-05	400	41.5	37.124
MI-06	500	47.14	40.846
MI-07	600	50.24	41.8564
MI-08	700	54.246	41.954
MI -09	800	56.54	42.554
MI-10	900	58.154	43.5124
MI-11	1000	59.854	44.876
MI-12	1100	60.56	45.514
MI-13	1200	59.156	47.547
MI-14	1300	61.134	47.8984
MI-15	1400	62.354	48.8452
MI-16	1500	64.17	48.99
MI-17	2000	66.249	51.846
MI-18	3000	67.6	52.44

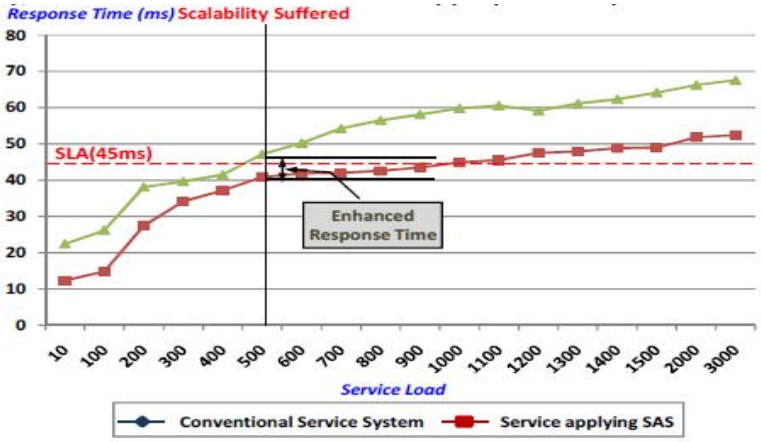

Figure 2. Chart of service migration result

6.2. Scalability of Web Applications in a Compute Cloud

In [4], the authors present the important scalability issue of performance indicators. They present a case study on the scalability and performance of web applications in the cloud. To illustrate the powerful scaling capabilities of cloud environments they introduce a novel scaling scenario for web applications deployed in virtual machines that are created and destroyed on demand.

To explore the key scaling indicators, they have carried out the performance measurements on an online collaboration web application. This web application is intended for different groups of enterprise users to share online business documents and to organize and track their organizational user contact information.

The performance measurements are mainly focused on monitoring the usage of system resources and access success and failure rates when large numbers of users simultaneously access web application. The cached session cookies corresponding to each logon user are then used subsequently to access different web pages in the web application.

The main problem with such web applications is the inability to plan ahead or even predict the number of users who will be accessing the applications. A solution is to scale the web application in a dynamic manner and let the number of web servers and web application components grow or shrink on demand.

They developed a web application for controlling the provisioning and de-provisioning of web-server VM instances, a dynamic scaling algorithm based on relevant threshold or scaling indicator of the web application. The scaling indicator that is selected here is the number of active sessions or logon sessions in each web application.

Based on the moving average of the scaling indicator, a dynamic scaling algorithm is used to trigger a scaling event to the provisioning subsystem. Depending on the updated statistics, action to scale up or down may be initiated. Scaling up or down means that an event will be triggered that instructs the provisioning subsystem to start or shut down web-server virtual machine instances in the cloud. When the web application is scaled up, the newly started virtual machine instance will run the web application. After the web application instances are ready, the front-end load-balancer configuration file is then updated and refreshed to place them into active services.
As mentioned previously, the scaling algorithm is implemented in the Service Monitor sub-system, and is used to control and trigger the scale-up or down in the Provisioning sub-system on the number of virtual machine instances based on the statistics of the scaling indicator. A hybrid approach is used to support both goals of resource maximization on individual virtual machine instances and minimization of total number of instances, in contrast to the typical approach of load balancing among available resources.

Figure 3 presents the experimental results on both the access failure rate and the access response time as a function of logon users with the implementation of our dynamic scaling algorithm on a Cloud. It is observed that there is no access failure (i.e. zero failure rate) even with more than 50,000 logon users to the web application (i.e. 10x more logon users than a single web application instance can support), and the access response times are kept relatively small and constant throughout the experiments. At one time under the peak load conditions, it is found that there are more than 12 virtual machine instances dynamically created and started in the Cloud.

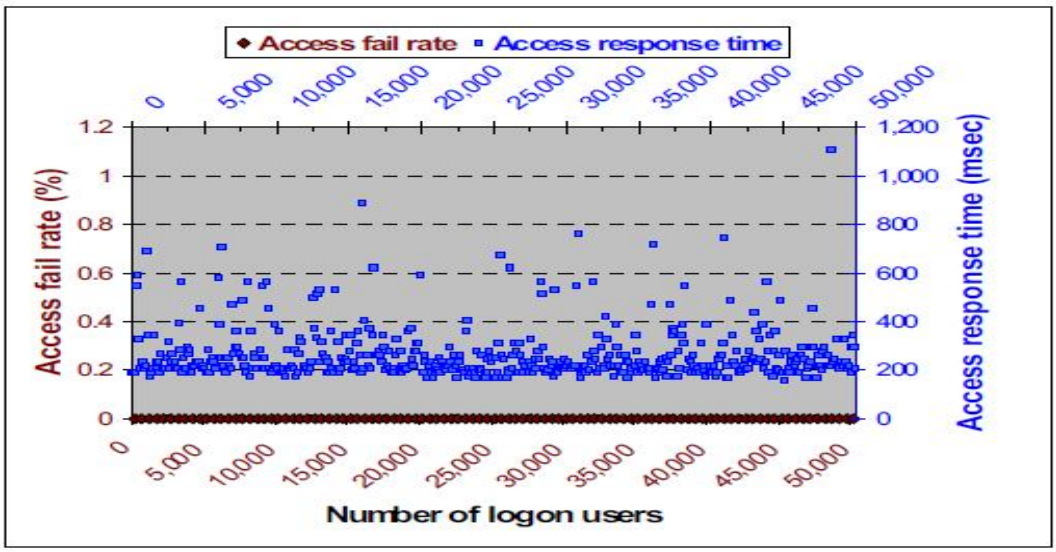

Figure 3. experimental results on access fail rate and access response time as a function of logon users with the implementation of the dynamic scaling algorithm on a cloud

6.3. Cloud Computing Infrastructure and Application Study

Ye & Qu have conducted research to detect the close relationship between IaaS and applications [8]. They propose a cloud-based infrastructure that is optimized so as to support large-scale agriculture information computing. This cloud infrastructure mainly consists of a virtualization platform for agriculture information Cloud Computing and management. At the same time, the research also provides insights about market-based resource management strategies that encompass both customer-drive service and management. Furthermore, the research evaluates the performances of CPU and Internet-based service workloads in the environment of the proposed Cloud Computing platform infrastructure and management service. Experiments show that the proposed Cloud Computing infrastructure and management service is effective and essential for large-scale agriculture information computing. Also, it has presented various cloud efforts in practice from a market-oriented perspective to reveal the emerging potential third-party services that enable the successful adoption of Cloud Computing.

The research indicate that management service platform furnish a overall management module. Cloud management platform can reflect working environment of cloud system, include environment of software and hardware. At the same time, cloud management platform can manage all resource of cloud computing platform and effectively assign it, include virtual machine.

Management service platform furnish a monitoring server. Figure 4 shows monitoring platform can monitor amount of total CPU, Hosts up and Hosts down. At the time, monitoring server can display the using instance of CPU, Memory and Nodes, and so on. We also can calculate average workload of CPU by this monitoring-platform.

Management service platform furnish a cloud computing center performance monitoring module. Figure 5 shows overall performance of the agriculture information cloud can be monitored. This monitoring platform can monitor overall workload of agriculture information cloud and real timely display in graphics interface. Once system manager monitor system workload is very scale, they can immediately deal with. At the time, we can improve the throughput of the agriculture information cloud.

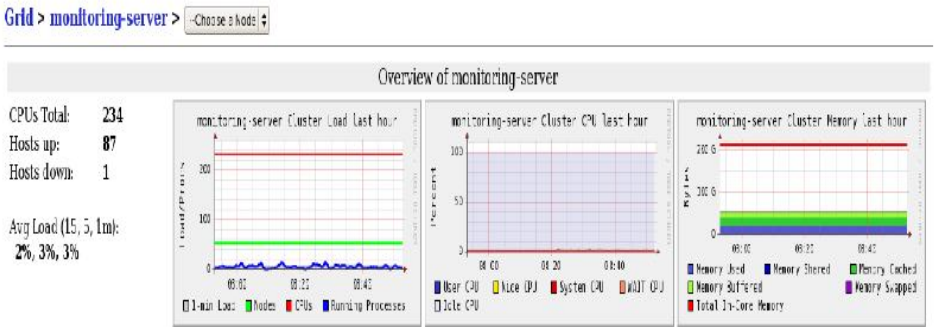

Figure 4. Service monitoring platform

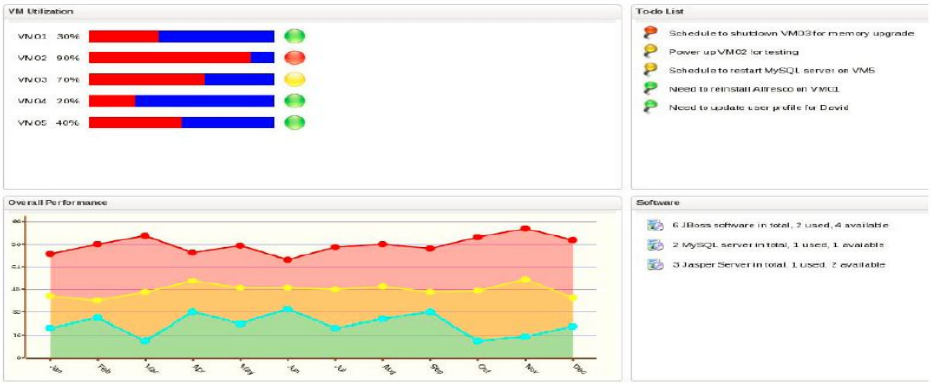

Figure 5.Cloud platform performance monitoring

In summary, all this researches clarifies that execution of the application on external resources cannot happened in an easy manner and performance plays a key role in scalability operations. They suggested a mechanisms and techniques to perform good quality of service to support moving to cloud computing.

7. Conclusion

Cloud computing is a recent technology trend that help companies in providing their services in a scalable manner. Hence, used this service capabilities required many procedures in order to get better performance.

References

[1] Z.Liu, "Research on Computer Network Technology Based on Cloud Computing," in Proceedings of the 9th International Symposium on Linear Drives for Industry Applications, Springer, 2014.
[2] J.Lee and S. Kim, "Software Approaches to Assuring High Scalability in Cloud Computing," in IEEE International Conference on E-Business Engineering, 2010.
[3] B.Furht and A.Escalante, in Hand Book of Cloud Computing, Springer, 2010.
[4] T.Chieu, A.Mohindra and A. Karve, "Scalability and Performance of Web Applications in a Compute Cloud," in e-Business Engineering (ICEBE), 2011.
[5] L.Vaquero, L.Rodero-Merino and R. Buyya, "Dynamically Scaling Applications in the Cloud," ACM SIGCOMM Computer Communication Review, pp. 45-52, January 2011.
[6] J.McCabe, Network Analysis,Architecture, and Design, Elsevier, 2007.
[7] C.Fehling, F.Leymann, R. Retter, W. Schupeck and P. Arbitter, Cloud Computing Patterns, Springer, 2014.
[8] M.Ye and Z. Qu, "Cloud Computing Infrastructure and Application Study," 2012.

[9] J.Dai, "Application of Cloud Computing in Campus Network Based on IaaS," in Recent Advances in Computer Science and Information Engineering, 2012.
[10] G.Reese, Cloud Application Architectures, O'Reilly Media, 2009.
[11] F.Galán, A. Sampaio, L. Rodero-Merino, I. Loy, V. Gil and L. Vaquero, "Service specification in cloud environments based on extensions to open standards," in Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middlewaRE, 2009.
[12] A.Younge, G.Laszewski, L. Wang, S. Alarcon and W. Carithers, "Efficient Resource Management for Cloud," 2010.
[13] M.Mollah, K.Islam and S. Islam, "Next Generation of Computing through Cloud Computing Technology," in 25th IEEE Canadian Conference on Electrical and Computer Engineering, 2012.
[14] M.Weinstein, "Planning Enterprise Networks to Meet Critical Business Needs," in Enterprise Networking Mini-Conference, 1997.
[15] F.Chanchary and S. Islam, "E-government Based on Cloud Computing with Rational Inference Agent," in High Capacity Optical Networks and Enabling Technologies (HONET), 2011.