Computer Science and Information Technology 1(1): 9-18, 2013 http://www.hrpub.org
DOI: 10.13189/csit.2013.010102

Dynamic Architectural Framework for Cloud Computing

Hassan Reza[*], Nitin Karodiya

School of Aerospace Sciences University of North Dakota Grand Forks, U.S.A.
*Corresponding Author: reza@aero.und.edu

Abstract Cloud Computing has been a buzzword for quite a while. A lot of companies are offering cloud infrastructures and services, which can be used by organizations and individuals at a nominal charge. There are various cloud providers in the market. But, when it comes to interoperability or quality attributes of the different types of cloud services by different cloud providers, there is no consensus on standards. The other but yet important issue is there is no framework, which can define the quality attributes of a cloud and measure them. Different companies have used different architectural patterns for implementing their cloud. But, none of them have proved to provide a good balance of the quality attributes such as Greenability, Availability, Security, Reliability, Performance, Portability or Interoperability etc. In this paper, we proposing an architectural framework for defining and measuring the quality attributes of a cloud and to customize these quality attributes to satisfy the quality requirements of different subscribers, which can make the cloud more usable and adoptable by reducing the cost and increasing the profit for the cloud vendors. It also makes it possible for the same cloud to behave differently for different subscribers according to their needs.

Keywords Cloud Computing, Software Architecture, Quality Attributes

1. Introduction

Cloud computing is an emerging field that is expected to change the entire information technology processes and IT market in future. It is changing how we develop, deploy, use the information technology and the infrastructure that it runs on. The cloud is typically built in a multilayered manner. Cloud is comprised of the infrastructures, which are used only on demand and these infrastructures are released after the completion of the task. Clouds can be classified in various types [35]: Public, Private, Hybrid and Virtual Private. The types of cloud are described in brief as follows:

Public Cloud – All the services and infrastructure are accessed through Internet and the vendor's premises host the cloud infrastructure. The hardware and software services are supplied as services upon demand. The cost is distributed across the businesses and hence reduces the capital cost.

Private Cloud – The infrastructure of the cloud model is implemented at the organizations data center and is dedicated to that particular organization, thereby suffering no bandwidth restriction or exposure to outside public. The service can be hosted by the organization or externally hosted, which is cheaper. External hosting can keep the product or service under strict control.

Virtual Private Cloud – It exists within a shared or public cloud.

Hybrid Cloud – It's the mixture of private clouds, for the secured application data and the public cloud for the less concerned application data that saves cost. It is the concept of abstraction of the interaction between the software (operating system) and hardware (servers). This improves speed, makes the system more flexible, reduces implementation costs, etc.

The cloud service is also classified based on the architecture. When the application itself is provided as a service over the Internet, it is called Software-as-a Service (SaaS). Cloud providers install and maintain the software on the cloud which the cloud users can use without the need to have the software installed on their local machines. The platform for developers to collaborate and interact is Platform-as-a-Service (PaaS). Cloud providers provide an entire computing platform, typically comprising of an operating system, development tools, databases, etc. Cloud providers provide enterprise users with hardware that can be implemented as data centers, saving them the cost of acquiring and implementing the hardware. This is called Hardware-as- a-service. Users are provided with resources ranging from computer hardware to virtual machines, storage, firewalls, etc. The utilities are provided on demand under Infrastructure-as-a Service (IaaS) [41].

Cloud computing has several benefits like reduced cost, increased speed and flexibility. Also, cloud computing not only provides low cost infrastructure and services but it also frees the clients from the trouble of installing and maintaining those infrastructure, services and applications in-house locally. The main features of cloud computing for the subscribers are on-demand availability of dynamic resources. But, like every new technology cloud computing also has some shortcomings and challenges. These challenges are concerned with the quality attributes of the cloud like Security and Privacy, Availability, Performance, Latency, Reliability, etc. Though the cloud providers guarantee to address these issues they don't take care of these issues completely. For e.g. Cloud providers say that their cloud is highly secure as well as highly available to the clients as per their requirements, but they are unable to fulfill these conflicting requirements. The cloud providers have to address these issues and challenges completely in order for the clouds to sustain. These are critical factors for the maturing of Cloud Computing technology and also for wide adoptability. Customers or clients still don't trust the clouds completely and hence in-spite of the research advances made in this area this is not such a mature technology in the market from the business point of view.

The main objective of this paper is to resolve the issues related to the quality attributes of the clouds, which conflicts with each other. For e.g., suppose there is a cloud provider whose cloud is highly available. Now the cloud provider is approached by a subscriber A whose primary requirement is the availability while he can be a little bit flexible about the security i.e. he can adjust with less secured cloud if he gets the required availability. Subscriber A will definitely subscribe to this cloud as he is getting availability, which is the primary focus of the cloud. Now suppose a new subscriber B comes to the cloud provider and asks for a highly secured cloud. As the primary focus of the cloud was availability it does not guarantee the level of security subscriber B wants. However, B is willing to compromise for availability for a more secured cloud. The cloud provider has only 2 choices now. Either he changes the architecture of the cloud and make it highly secure which will cost him a lot and also there is a risk of losing subscriber A as his requirement of availability will not be satisfied with the architectural change of the cloud or he may loose subscriber B by saying that the cloud is unable to provide the security required by subscriber B which will cost him in terms of business. Clearly, none of these choices are good for the cloud provider and for the sustainability of the cloud. This paper resolves this dilemma on part of the cloud providers and answers the problem related to these conflicting requirements.

In this paper we are proposing an architectural framework, which can be adjusted according to the needs of the customers to focus on the quality attribute that they desire and increase the sustainability of the cloud even if these quality attributes requirements from different customers conflicts each other. This framework guarantees the specific quality attributes to the subscribers of the cloud by using specific algorithms based on the properties specified in the subscribers account.

This paper is organized in separate sections. After introduction section there is a section for related work, which is then followed by approach. The approach section talks about the issues related to the clouds and my proposed architectural framework. Next is the methodology section, which discusses how this architectural framework will achieve the specific results. The methodology section also discusses some hypothetical case study and thus explains how this architectural framework will achieve these results and how this will be beneficial. The methodology section is followed by the conclusions and future work section.

2. Related Works

There has been a lot of research in the Cloud Computing lately. Various researchers have addressed issues related to the quality attributes in the clouds. To tackle some of these issues"Service Oriented Cloud Computing Architecture" (SOCCA) can be used [1,2]. The main multitenant pattern within SOCCA is Single Application Instance and Multiple Service Instances (SAIMSI). This pattern not only provides better scalability but also easy customizability. The other multitenant patterns are Multiple Application Instance and Single Application Instance which are described in [3]. SOCCA supports both SOA and Cloud computing which complements each other. SOCCA supports easy migration of application from one cloud environment to another by promoting open standards.

In [6] an attempt has been made to specify the requirements for better cloud environment using *Open Virtualization Format* (OVF). In addition to the extensions to the OVF syntax that supports clouds including attributes for cross virtual machine reference [7], etc., they added abstractions to OVF in the form of rules that allows the conditions related to the application and should be met with the *application domain description*. [39] Focus is to compare the quality information depending on cloud service type based on content analysis and Cloud Service Quality Model (CSQM). CSQM supports communication between the service providers and the service consumers. It has five quality factors: Service Level Quality, Integration Quality, Security Quality, Usage manageability and Metering Quality. Usage manageability and Meter Quality are grouped to Manageability Quality group and the rest to service quality group. The analysis is done using these quality factors but it only provides the details about the quality of cloud service. It doesn't provide a solution to maximize the quality or to improve the certain quality factors based on the subscriber's requirement.

The market-oriented resource management architecture [35] provides the different needs and QoS to the customers. It gives a practical standpoint on how the clouds should be implemented to give maximum benefit to the customers and providers of the cloud from the business point of view but it fails to address various issues related to cloud quality attributes and also does not give the techniques that can satisfy specific quality needs of the individual subscribers.

In [26] the authors have described a novel PaaS architecture, which guarantees the real-time Quality of Service. QoS parameters are given specific details at both application and infrastructure details in this work. To provide these features the PaaS architecture consists of two key elements QoS Service Oriented Engineering and

On-demand Service Management. The proposed referenced architectural model focuses on decoupling of physical resource management from virtual resource management and capability to mediate between services and computing resources dynamically in real-time. This architecture contains FCAPS based (Fault, Configuration, Accounting, Performance and Security) service mediation, which enables the global management of computing resources irrespective of their geographical locations. This management platform provides a pool of physical computing resources i.e. network, storage, processors, etc. that can be organized dynamically in real-time to form a cloud computer and this cloud computer can grow or shrink based on the demand in real time while providing the desired level of security, performance availability and other quality attributes for the clouds. This architecture will help the cloud providers, service developers and end users in providing real-time dynamism of infrastructure and services and will serve as a next generation cloud architecture. But the main concern with this approach is that decoupling cannot be fully achieved which limits the application of this architecture to the clouds.

In this work[28], the authors have proposed a novel architecture for cloud storage, which is layered and cooperative. They have also discussed the key technologies such as deployment, storage virtualization, security, data migration and organization, etc. In [40], Sector is proposed which enables the users to work with large datasets stored on different machines across the network as if they are local. Meta CDN is introduced in [30] that creates an integrated overlay network to pro- vide high performance CDN to content creators at low cost. [31, 32] describes a live data migration mechanism for storage of data in clouds. The authors further discussed the key technologies of the cloud storage servers such as deployment, virtualization, availability, data organization, etc. of cloud storage. In addition to these they also discussed data migration and load balance, data de-duplication [33,34], and storage security. The authors also discussed the operation mechanism for the storage architecture and key technologies of the clouds. The operation mechanism focuses on dynamic and live data organization, service delivery and evolution, data security etc.

Cloud computing services can be used to lower the overall cost of ownership but still many organizations prefer their internal existing IT systems than the cloud-based systems due to the security issue in clouds [8, 9]. The design of Trusted Cloud Computing Platform (TCCP) [7] is another approach to address the security issue in clouds. TCCP provides a closed computing/executing environment via Infrastructure as a Service (IaaS), which allows the confidential execution of virtual machines in clouds. It also enables the user to verify whether the cloud service is secure or not before the users start their virtual machine execution. The Trusted platform [10,11,12,13] uses the features of the Trusted Platforms Module (TPM) chips for attesting the remote machines. TCCP extends the concept of TCM and takes it to the IaaS backend, which provides a closed box execution environment for users VM on all hosts.

The performance is measured based on the successful responses rate for all the requests made [29, 30]. Increasing the replication can increase this. Availability, reliability, safety, integrity and maintainability are attributes of dependability, which can be attained through fault prevention, fault tolerance, fault removal and fault forecasting. Based on average load balancing performance is measured and SLA negotiation and enforcement can be done [30].

Cloud management is responsible for all computing re-sources of the cloud and the primary driving force for cloud computing is the global resource optimization. This management of resources depends on virtualization in cloud environment [16]. Each application sees a virtual environment completely dedicated to it and the cloud management allocates the share of resources to these applications. The main objective of cloud providers is to optimize the use of resources while providing the quality of service, which is measured in terms of response time by the authors in this article [15]. The authors have shown how the approach for resource management described in [17,18,19,20] can be applied in cloud computing. In this approach, they have used a performance model to combine the system features and issues. This approach tracks the model and optimizes the resource management using the model. The performance model is used to predict the outcomes of changes on various decision parameters. The complex interactions between these decision parameters are very difficult to predict without a model. The LQM [21,22,23] performance model is used because of its correspondence to the architecture and its layered representation for resource behavior [24]. The optimization technique used by authors is based on a network flow model (NFM [25]). The approach provides very important quality attribute for the cloud (i.e. Greenability) as it optimizes the resource utilization thereby reducing the CO2 emission.

Another approach for optimal resource usage is to make every service attempt to optimize itself locally [5]. Every service will look for other services that can offer it the attribute it wants to optimize [29]. For e.g.- if the web service wants to optimize performance, it will negotiate a SLA with another service that provides higher bandwidth for speed or more efficient algorithm for faster processing. If the service wants availability, it will create an SLA with another service that provides redundancy and backup, so in case of high traffic, the web service will be able to scale up accordingly. At the same time, when traffic is slow, the web service can switch to other services providing lower bandwidth, as speed is no longer a necessity. As long as the quality of service specified in the contract with the user business is upheld, it doesn’t matter what components are used, but this approach helps the clouds perform optimally and to focuses on the management and performance of the resources. The resources can be process memory, storage and, network availability. [29,30] takes in consideration of the performance evaluations that were done on grid. The important issue grid had is, to manage, allocate and reserve resources for the successful completion of the scheduled tasks. The performance of the system is evaluated just not by the amount of work done but the work done in comparison with the time and resource used. Fault tolerance is an important factor to improve the performance. The scheduled tasks have to be monitored and if the task fails due to resource unavailable, then the task should be restarted with the alternate available resource. The tasks should be check pointed for fault tolerance to function. The resource management is one important technique that differentiates grid from cloud. The current trend to cloud is to virtualize the resources at the lower level. The technology to do that is Hypervisor that is a Virtual machine Manager [13, 14].

In general, there are six types of virtualization. Full virtualization, hardware assisted virtualization, partial virtualization, para virtualization, hybrid virtualization, and operation system-level virtualization. The disadvantage of using virtualization is that the guest machine and the host machine should have the same Operating System. And when the research proceeds it is very clear that the emphasis is on the efficient distribution and replication of data geographically.

In [38] the authors proposed a cloud model, composed of three delivery models and four deployment models, to promote availability. The security and privacy of the data are crucial needs and they are the responsibilities of both the cloud providers and the customer or subscriber. But not all non-functional performance can be measured, such as privacy and security. The cloud faces the same issues as any traditional system like insecure user authentication, weak password, information leakage, hijacking, data loss and many others [41]. Role based access Control (RBAC) has benefits like simplicity, flexibility and efficiency [38]. The primary rules defined by it are role assignment, role authorization and permission authorization. This technique can be further extended to credential based RBAC which specifies who can access and from where and when they can access. Also, to increase the security, the data stored in the cloud can be encrypted. Homomorphic encryption is one of the proposed encryption techniques, which lets meaningful computation on the encrypted data, and identity based encryption uses the public key approach to encrypt and hence has a non-interactive authentication frame-work [39]. This technique also incorporates features like automatic expiration and unread ability after certain date that enhances the security [39]. But the centralized storage of public key generator requires more security and there is a requirement of secured channel to send the private key.

Though, the emergence of cloud computing has combined the benefits of grid computing, utility and autonomous computing but the traditional security mechanisms are inefficient for the information and computing security in the clouds [36]. In [27], some security requirements for all service levels (i.e. SaaS, PaaS, IaaS), and physical level security of data centers have been identified. They are: (i) Privacy in multitenant environment (ii) Data Protection from Exposure (iii) Access Control (iv) Communication Protection (v) Software security (vi) Service Availability (vii) Data Security (viii) Security of Cloud Management control (ix) Secure virtual machines (x) Hardware security (xi) Hardware reliability (xii) Network Protection, etc. To satisfy these security requirements the authors propose the establishment of Trusted Third Party (TTP) in the clouds. TTPs form a network of Public Key Infrastructure (PKI) as they are connected through the chains of trusted paths [27].

A combination of PKI, LDAP and Single-Sign On (SSO) can mitigate against majority of the threats, and hence provides security in clouds, which is very important quality attribute. The related works we have discussed in this area address one or more particular issue, but they do no take into consideration the quality attribute requirements of the customers. In this paper, we have made an attempt to address the issues related to quality attributes of the clouds while considering the requirements of the customers.

3. Proposed Method

Various architectures are used for increasing the quality of cloud by leveraging one particular quality attribute, which is fixed for particular cloud environment. But when another quality attribute is of most priority for some particular customer it is not possible to customize the cloud for this new customer. For example, suppose a cloud provider has built the cloud to be highly available. Then the customers or the cloud subscribers who demand high availability will be satisfied with this cloud. Now, let’s suppose a new customer demands a highly secure cloud environment. Then this new customer’s primary requirement (i.e. high security) is not fulfilled by the mentioned cloud architecture. Therefore, the new customer will most likely not subscribe to this cloud provider, which is business loss to the provider. Profitability is the main goal of any cloud provider who also wants maintaining acceptable level of cloud services and providing low cost infrastructure without the degradation in service thereby making cloud computing a win-win situation for both cloud providers and cloud users.

The architectural framework discussed here will make cloud computing a win-win situation for both cloud providers and users. This architectural framework works with the existing clouds and turns them into a highly effective and efficient cloud which satisfies the specific quality requirements of the subscribers without increasing the cost. This framework makes use of the hypervisor in the cloud that starts up the virtual machine based on the traffic or computing demands.

The hypervisor of the cloud is responsible to start the virtual machines for computing or for meeting the demands for computational resources in the cloud. The hypervisor assures that every subscriber’s data or applications are isolated from each other. It does so by starting the virtual machines according to a particular algorithm to manage resources effectively while meeting the demands of the subscribers. The proposed architectural framework asks for a slight change in the algorithm of the hypervisor to consider the specific quality attribute requirements of the subscribers. The approach asks for gathering the quality attribute requirements of the customers and set this as a property in the property file for the particular subscribers account. The various nodes in the cloud follow specific architectural pattern and specific algorithm and serve the subscribers. Every node in the cloud is aware of various algorithms in the cloud to satisfy the quality attribute requirements of particular subscriber. When the demand for resources increases in the cloud, the hypervisor checks the different accounts and find out which subscriber has asked for more resources. After getting the identity of the subscriber the hypervisor reads the property file and find out the specific quality requirements from that file. Once known the quality requirement the hypervisor starts up the virtual machines and ask them to operate in a particular mode according to the algorithm that is designed to satisfy the quality requirements. Basically, it starts the virtual machines or nodes that behave in a certain way according to the algorithm, which meets the specific quality requirements as specified in the property file for subscriber's account. Thus even when the cloud initially had focus on a particular quality attribute, it still can satisfy other quality attributes up to the desired level. This approach works even when the quality attributes are conflicting.

The proposed architectural framework is generic and can be applied to any cloud and can be utilized to satisfy any of the quality attributes as desired by the subscriber. It is a dynamic framework that enhances the quality attributes as per the requirements and thereby making it possible for the same cloud to behave differently to different subscribers according to their needs. It increases the sustainability of the clouds by attracting more and subscribers to the cloud there by keeping the high profit for the cloud providers and low cost for the subscribers.

3.1. Methodology: Proposed Architectural Framework

This section will discuss how the proposed architectural framework will achieve its purpose of satisfying conflicting quality attributes for different subscribers with the help of a hypothetical case study. It will give a comparison of current approach and the approach using the proposed architectural framework when different subscribers have conflicting requirements.

(i) Current methodology without the proposed architectural framework

Any cloud, which was built with focus on a quality attribute Q exists. A potential subscriber A approaches the cloud vendor. Subscriber A wants the specific quality attribute Q (e.g., Scalability) to be satisfied according to his requirements. Since cloud vendor provides quality attribute Q, the potential subscriber subscribes to the cloud. Both the cloud provider and cloud subscriber are happy. Now suppose another potential subscriber B approaches the cloud vendor and asks for the quality attribute P, which the cloud doesn't provide currently. Moreover, the quality attribute conflicts with the quality attribute Q. Now there are two possibilities for the cloud vendor. First, it refuses to B by saying that it cannot provide the quality attribute P desired by B, which will cause a loss in business. Second, the cloud vendor agrees to provide the quality attribute P to the subscriber B and thereby changes the architecture, which will cost him money and also there is a risk of losing subscriber A because the architectural changes of cloud might affect the quality attribute desired by A. In this case, the cloud vendor is again risking losing money. So, both the choices will affect the business and also cost money to the cloud provider. This is graphically modeled in the Figure 1.

(ii) Methodology using the proposed architectural framework

Any cloud, which was built with focus on a quality attribute Q exists. A potential subscriber A comes to the cloud vendor. A want the specific quality attribute Q to be satisfied according to his requirements. Since cloud vendor provides quality attribute Q, the potential subscriber subscribes to the cloud. Both the cloud provider and cloud subscriber are happy. Now suppose another potential subscriber B comes to the cloud vendor and asks for the quality attribute P, which the cloud doesn't provide currently. Moreover, the quality attribute conflicts with the quality attribute Q. Now the cloud provider uses the proposed architectural framework and the quality attribute requirements are specified as property for the subscribers account. Now, when the subscriber demands for the resources, only the nodes, which satisfy the property and follows the particular algorithm for maintaining the quality attribute defined as the property for subscriber B will be selected and a cluster for subscriber B will be made out of those nodes. In this way the requirements of B as well as A are satisfied and the cloud behaves like it is specifically satisfying the requirements of a particular subscriber but in fact it behaves differently for different subscribers to satisfy their quality attribute requirements. So, both the subscribers as well as cloud provider are happy and it is a win-win situation for all of them. This is graphically modeled in the diagram below:

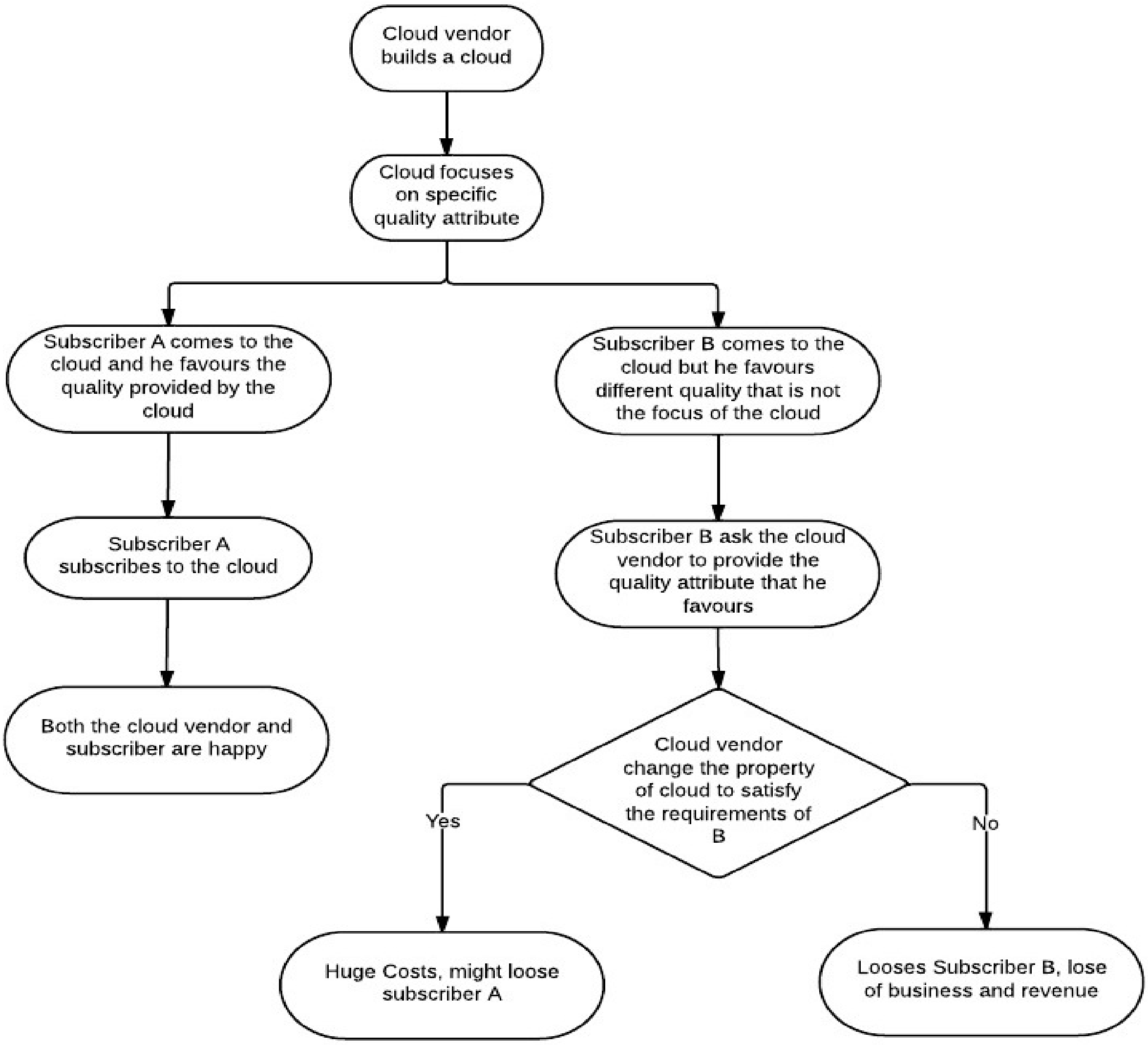
Cloud vendor builds a cloud
Cloud focuses on specific quality attribute
Subscriber A comes to the cloud and he favours the quality provided by the cloud
Subscriber B comes to the cloud but he favours different quality that is not the focus of the cloud
Subscriber A subscribes to the cloud
Subscriber B ask the cloud vendor to provide the quality attribute that he favours
Both the cloud vendor and subscriber are happy
Cloud vendor change the property of cloud to satisfy the requirements of B
Yes
No
Huge Costs, might loose subscriber A
Looses Subscriber B, lose of business and revenue

Figure 1. Proposed Architectural Framework Satisfying Cloud Requirements

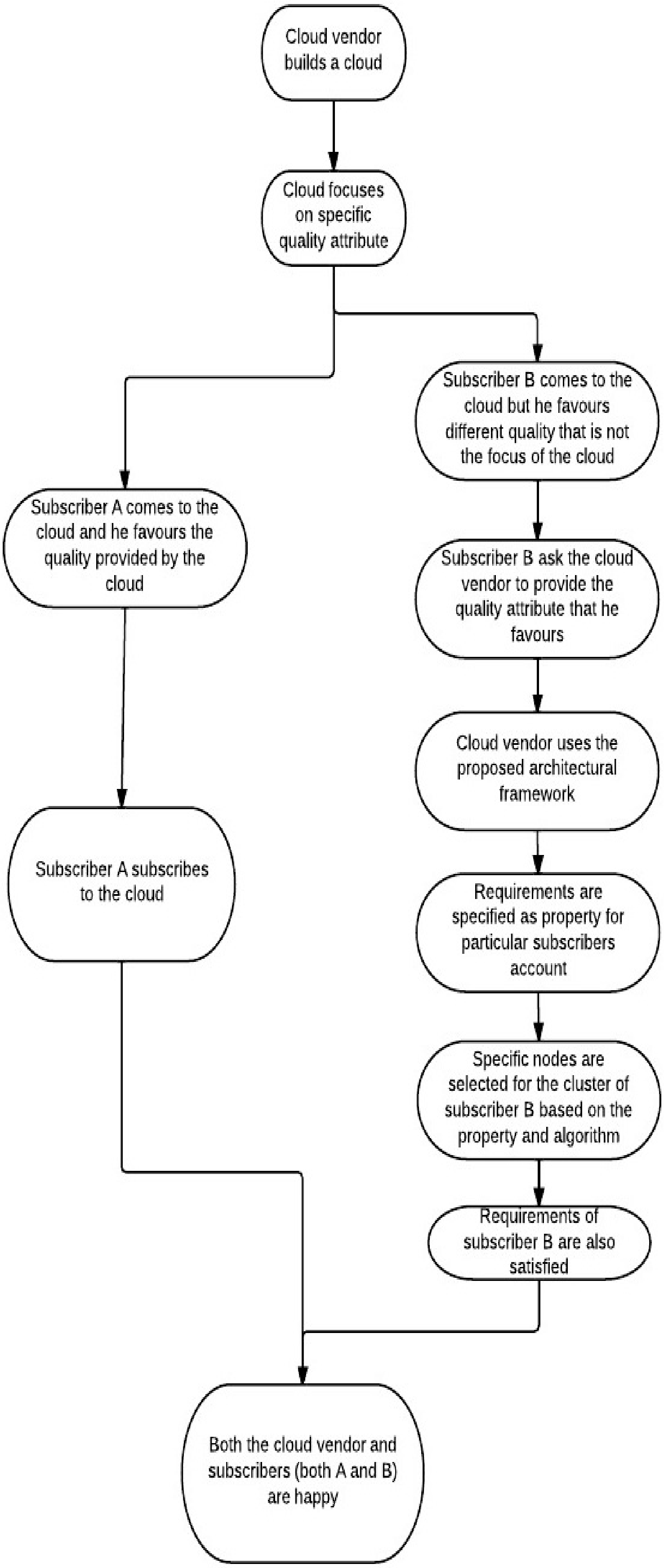
Cloud vendor builds a cloud
Cloud focuses on specific quality attribute
Subscriber A comes to the cloud and he favours the quality provided by the cloud
Subscriber B comes to the cloud but he favours different quality that is not the focus of the cloud
Subscriber B ask the cloud vendor to provide the quality attribute that he favours
Cloud vendor uses the proposed architectural framework
Subscriber A subscribes to the cloud
Requirements are specified as property for particular subscribers account
Specific nodes are selected for the cluster of subscriber B based on the property and algorithm
Requirements of subscriber B are also satisfied
Both the cloud vendor and subscribers (both A and B) are happy

Figure 2. Win-win Framework Satisfying Both Vendor and Cloud Customer

4. Hypothetical Case Study

Assume that a cloud exists that has 90% Security and 90% availability. Both availability and security are conflicting requirements. There are two subscribers A and B of the cloud. A desires at least 99% availability and it doesn't care about the security i.e. he is willing to compromise security for getting desired availability as its application doesn't uses any confidential data which can be stolen or modified etc. For example, A wants to transfer its CRM application on cloud. B desires at least 99% security and it doesn't care about the availability i.e. it is willing to compromise a little bit of availability for the desired security. For e.g. B have an application that requires health records and other personal details of any individual. Clearly, the examples of applications given for A and B care more about availability and security respectively.

According to the proposed framework their quality attribute requirements are specified in the property file for the subscribers' accounts. Figure 2 shows this process. When the cluster or virtualization for A is formed, the hypervisor's algorithm selects all the available nodes and clone the data and application of A on them so that it fulfills the availability requirements of A. When the cluster or virtualization for B is formed, the hypervisor's algorithm selects the nodes which follows an specific algorithm such that it forms a specific mini cloud inside the cloud and thus improves security thereby fulfilling B's requirement.

Thus the cloud behaved differently for different subscribers and fulfilled the quality attribute requirements for both the subscribers although their requirements were conflicting. The figure 3 shows the image of cloud at this point of time.

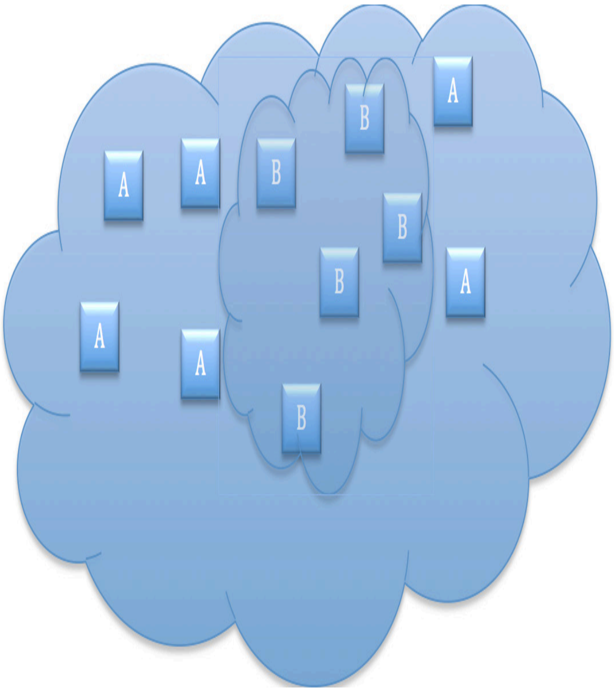

Figure 3. Nested Clouds

5. Conclusion and Future Work

In this paper we have presented a potential dynamic architectural framework for clouds that can enhance the quality attributes as per requirement of the users. At first we have discussed the existing issues in the cloud computing area, we then have discussed the specific issues that our proposed framework will address. In our proposed approach, we have discussed how this framework can be implemented. To this end, we provided the details needed to understand this framework. To show the feasibility our work, a case study using our proposed architectural framework was shown. The main advantage of our proposed architectural framework is that this framework can be applied to existing clouds, and it works well with conflicting quality attribute requirements relevant to the cloud computing. Examples of these quality attributes are security and performance, scalability and performance, etc. Another immediate benefit of our proposed framework is that it makes the maintenance of the cloud easy as the requirements change, because only the property for the particular account and algorithm needs to be updated. It also makes it easier for the subscribers to chose the quality attributes required by them and hence allows them to pay only for those quality attribute requirements needed at any specific time and not for the ones which are not needed and are provided by the cloud. Therefore, the other benefit of results of this framework is that it should also improve sustainability of the clouds as it is helping in maintaining the low cost, on demand dynamically available resources for computing.

Though this proposed framework has various advantages as mentioned, but it needs to be tested and verified by implementing the solution to a real cloud. Once this framework is verified for the results using the cloud logs, this framework can be extended for hybrid clouds, which are a little bit more complex in terms of their operation and maintenance than the public clouds.

REFERENCES

[1] Tsai, Wei-Tek, Xin Sun, and JanakaBalasooriya. ”Service-oriented cloud computing architecture.” Information Technology: New Generations (ITNG), 2010 Seventh International Conference on. IEEE, 2010.

[2] Tsai, Wei-Tek, et al. ”Service-oriented user interface modeling and com- position.” e-Business Engineering, 2008. ICEBE'08. IEEE International Conference on. IEEE, 2008.

[3] Huang, Ying, et al. ”Framework for building a low-cost, scalable, and secured platform for web-delivered business services.” IBM Journal of Research and Development 54.6 (2010): 4-1.

[4] Chapman, Clovis, et al. ”Software architecture definition for on-demand cloud provisioning.” Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing. ACM, 2010.

[5] Fox, Armando, and R. Griffith. ”Above the clouds: A Berkeley view of cloud computing.” Dept. Electrical Eng. and Computer. Sciences, University of California, Berkeley, Tech. Rep. UCB/EECS 28 (2009).

[6] Open Virtualization Format Specification. Specification DSP0243 v1.0.0, Distributed Management Task Force, 2009.

[7] Galn, Fermn, et al. ”Service specification in cloud environments based on extensions to open standards.” Proceedings of the fourth international ICST conference on communication system software and middleware. ACM, 2009.

[8] Santos, Nuno, Krishna P. Gummadi, and Rodrigo Rodrigues. ”Towards trusted cloud computing.” Proceedings of the 2009 conference on Hot topics in cloud computing. USENIX Association, 2009.

[9] Survey: Cloud Computing ’No Hype’, But Fear of Security and Control Slowing Adoption.

[10] Perez, Ronald, Reiner Sailer, and Leendert van Doorn. “vTPM: virtualizing the trusted platform module.” Proc. 15th Conf. on USENIX Security Symposium. 2006.

[11] Garfinkel,Tal,et al. “Terra:A virtual machine based platform for trusted computing.” ACM SIGOPS Operating Systems Review. Vol. 37. No. 5. ACM, 2003.

[12] Murray, Derek Gordon, Grzegorz Milos, and Steven Hand. ”Improving Xen security through disaggregation.” Proceedings of the fourth ACM SIGPLAN/SIGOPS international conference on Virtual execution environments. ACM, 2008.

[13] Sailer, Reiner, et al. ”Building a MAC-based security architecture for the Xen open-source hypervisor.” Computer Security Applications Conference, 21st Annual. IEEE, 2005.

[14] Sarathy, Vijay, Purnendu Narayan, and Rao Mikkilineni. ”Next Generation Cloud Computing Architecture: Enabling Real-Time Dynamism for Shared Distributed Physical Infrastructure.” Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), 2010 19th IEEE International Workshop on. IEEE, 2010.

[15] Li,Jim,etal.”Performance model driven QoS guarantees and optimization in clouds.” Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing. IEEE Computer Society, 2009.

[16] Steinder, Malgorzata, et al. ”Server virtualization in autonomic management of heterogeneous workloads.” Integrated Network Management, 2007. IM’07. 10th IFIP/IEEE International Symposium on. IEEE, 2007.

[17] Woodside, Murray, et al. ”Performance by unified model analysis (PUMA).”Proceedings of the 5th international workshop on Software and performance. ACM, 2005.

[18] Zheng, Tao, C. M. Woodside, and Marin Litoiu. ”Performance model estimation and tracking using optimal filters.” Software Engineering, IEEE Transactions on34.3 (2008): 391-406.

[19] Zheng, Tao, et al. ”Tracking time-varying parameters in software systems with extended Kalman filters.” Proceedings of the 2005 conference of the Centre for Advanced Studies on Collaborative research. IBM Press, 2005.

[20] Zheng, Tao. Model-based dynamic resource management for multi tier information systems. Vol. 68. No. 11. 2007.

[21] Franks, Greg, et al. ”Enhanced modeling and solution of layered queuing networks.” Software Engineering, IEEE Transactions on 35.2 (2009): 148-161.

[22] Rolia, Jerome A., and Kenneth C. Sevcik. ”The method of layers.” Software Engineering, IEEE Transactions on 21.8 (1995): 689-700.

[23] Woodside, M., et al. ”The stochastic rendezvous network model for performance of synchronous client-server-like distributed software.” Computers, IEEE Transactions on 44.1 (1995): 20-34.

[24] Franks, Greg, et al. ”Layered bottlenecks and their mitigation.” Quantitative Evaluation of Systems, 2006. QEST 2006. Third International Conference on. IEEE, 2006.

[25] Chinneck, John W. ”Processing network models of energy/environment systems.” Computers and industrial engineering 28.1 (1995): 179-189.

[26] Boniface, Michael, et al. ”Platform-as-a-service architecture for real- time quality of service management in clouds.” Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on IEEE, 2010.

[27] Zissis, Dimitrios, and Dimitrios Lekkas. ”Addressing cloud computing security issues.” Future Generation Computer Systems 28.3 (2012): 583-592.

[28] Zeng, Wenying, et al. ”Research on cloud storage architecture and key technologies.” Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human. ACM, 2009.

[29] Grossman, Robert L., et al. ”Compute and storage clouds using wide area high performance networks.” Future Generation Computer Systems 25.2 (2009): 179-183.

[30] Broberg, James, Rajkumar Buyya, and Zahir Tari. ”MetaCDN: Harnessing Storage Clouds for high performance content delivery.” Journal of Network and Computer Applications 32.5 (2009): 1012-1022.

[31] Hirofuchi, Takahiro, et al. ”A live storage migration mechanism over wan and its performance evaluation.” Proceedings of the 3rd international workshop on Virtualization technologies in distributed computing. ACM, 2009.

[32] Ying, Zhan, and Sun Yong. ”Cloud storage management technology.” Information and Computing Science, 2009. ICIC’09. Second Inter- national Conference on. Vol. 1. IEEE, 2009.

[33] Mandagere, Nagapramod, et al. ”Demystifying data duplications. “Proceedings of the ACM/IFIP/USENIX Middleware’08 Conference Companion. ACM, 2008.

[34] Storer, Mark W., et al. ”Secure data reduplication.” Proceedings of the 4th ACM international workshop on Storage security and survivability. ACM, 2008.

[35] Buyya, Rajkumar, Chee Shin Yeo, and Srikumar Venugopal. ”Market- oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities.” High Performance Computing and Communications, 2008. HPCC’08. 10th IEEE International

Conference on. IEEE, 2008.

[36] Buyya, Rajkumar, DavidAbramson, and Srikumar Venugopal. “The grid economy.” Proceedings of the IEEE 93.3 (2005): 698-714.

[37] Kim, J. K., and Jung-Hee Jeong. "Comparison of Cloud Service Quality Information Publication Based on Cloud Service Quality Model." 2012 International Conference on Information and Computer Applications (ICICA 2012), Singapore. Vol. 24. 2012.

[38] Takabi, Hassan, James BD Joshi, and Gail Joon Ahn. "Security and privacy challenges in cloud computing environments." Security & Privacy, IEEE 8.6 (2010): 24-31.

[39] Kumar, Ashish. "World of Cloud Computing & Security." International Journal of Cloud Computing and Services Science (IJ-CLOSER) 1.2 (2012): 53-58.

[40] Yunhong Gu, Robert L. 2009. Grossman. Sector: A high performance wide area community data storage and sharing system. Future Generation Computer Systems, 20 May 2009.

[41] Rimal, Bhaskar Prasad, Eunmi Choi, and Ian Lumb. "A taxonomy and survey of cloud computing systems." INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on IEEE, 2009.