Services Quality Improvement through Control Management Cloud-Based SLA

Abel Adane 1†
abeladi6@gmail.com
Arba Minch University

Abstract
Cloud-based technology is used in different organizations around the world for various purposes. Using this technology, the service providers provide the service mainly SaaS, PaaS and while the cloud service consumer consumes the services by paying for the service they used or accessed by the principle of "pay per use". The customer of the services can get any services being at different places or locations using different machines or electronic devices. Under the conditions of being well organized and having all necessary infrastructures, the services can be accessed suitably. The identified problem in this study is that cloud providers control and monitor the system or tools by ignoring the calculation and consideration of various faults made from the cloud provider side during service delivery. There are currently problems with ignoring the consumer or client during the monitoring and mentoring system for cloud services consumed at the customer or client level by SLA provisions. The new framework was developed to address the above-mentioned problems. The framework was developed as a unified modeling language. Eight basic components are used to develop the framework. For this research, the researcher developed a prototype by using a selected cloud tool to simulate and java programming language to write a code as well as MySQL to store data during SLA. The researcher used different criteria to validate the developed framework i.e. to validate SLA that is concerned with a cloud service provider, validate what happened when the request from the client-side is less than what is specified in SLA and above what is specified in SLA as well as implementing the monitoring mechanism using the developed Monitoring component. The researcher observed that with the 1st and 3rd criteria the service level agreement was violated and this indicated that if the Service level agreement is monitored or managed only by cloud service prover, there is a violation of LSA. Therefore, the researcher recommended that the service level agreement be managed by both cloud service providers and service consumers in the cloud computing environment.
Keywords:
Cloud Computing, Cloud Service Consumer, Cloud Service Provider, Service Level Agreement

1. Introduction

Currently, communication is the backbone of every organization that performs its operations in day-to-day activities. Information communication plays an important role in the development of every country. Cloud computing services are simple, user-friendly, and convenient to access at any time, from any location, and on any device. Several studies and case studies have revealed a recurring issue: monopolization of measuring and metering of consumer billing for Cloud Services consumed [1].Now a day the world is adopting cloud computing technology to provide and consume cloud-based services. According to NIST [2], cloud computing is defined as "a model for enabling universal, suitable, on-demand network access to a shared pool of configurable computing resources." Cloud computing has very important features that make cloud service simple, suitable, on-demand self-service, broad network access, and resource pooling [3-7].

Cloud computing services are delivered through software as a service, platform as a service, and infrastructure as a service delivery models. The main important characteristic of cloud computing services is Pay peruse, we only pay for the services we used and this makes the cloud acceptable and spread over the world. However, some cloud providers are not always accountable for the services they promised to provide to end-users under service level agreement [1,7]. Service level agreements are the guarantee of both cloud service providers and cloud service consumers but Cloud Service Providers (CSPs) have hidden mentoring tools, ignoring the calculation and consideration of service defects such as downtime, service outage, poor performance, and client service migration delays during server overloads.

Currently, there is a lack of a parallel monitoring and metering system for cloud services consumed at the customer level under the provisions of SLAs. The Cloud Governance Tier has not provided any flexibility or utility for monitoring such consumed cloud services at the client tier alongside the CSP tier [7,8]. Notably, In the cloud computing environment, there are three major components (Figure 1). The first layer of this component is the consumer or broker of cloud services. The second layer consists of applications such as web, mobile, desktop, or laptop computers that are used to request services from the cloud service provider. The third layer includes three critical subcomponents: service request

Manuscript received May 5, 2023
Manuscript revised May 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.5.11

examiner, SLA management & resource allocation, and cloud service provider [1, 6, 8].

2. Literature Review

More cloud service providers are offering services and infrastructures that were previously restricted to a single provider's data centers [9,10]. There are many studies conducted on the cloud-based service level agreement, but nobody of the researchers focused on monitoring the service level agreement from both the service providers and services consumer side of the services [8,11]. Cloud services are all services that are related to applications over the internet and hardware and software system in the data center that provides these services [11]. The term "private cloud" refers to internal data centers of a company or other organization that is not publicly accessible [8,12]. All the services delivered from the cloud service providers should be based on the agreement stated and the er service level agreements and are used as a guarantee for cloud service providers and cloud service consumers. When a service level agreement is violated, different kinds of problems will occur. Services will delay, downtime, and overloaded and as a result, the cloud consumers will be challenged to continue with the cloud services providers [13,14]. System administrators are lined from using sensitive information technology resources through the use of cryptography, authentication, and other techniques, and backup and log data are protected through the separation of administrator responsibilities [4,11]. Cloud Computing technology has a wide range of scales in which services are delivered to consumers through Internet connection services. In cloud computing, central remote servers and the Internet are used to host applications and data [1]. Entrust data, computation, and software from a user to remote services.

Cloud computing and storage solutions enable users and businesses to store and process data in third-party data centers in a variety of ways. It depends on resource sharing to achieve network coherence and economies of scale. Because of the low service cost, high-performance computing power, scalability, availability, and accessibility consumers are benefited from cloud computing technology [15,16]. Different scholars attempted to develop a client-side monitoring system for the greenness of cloud services (Figure 1), but this research focuses on both client- and server-side monitoring and control for all services, rather than just green services [12,15].

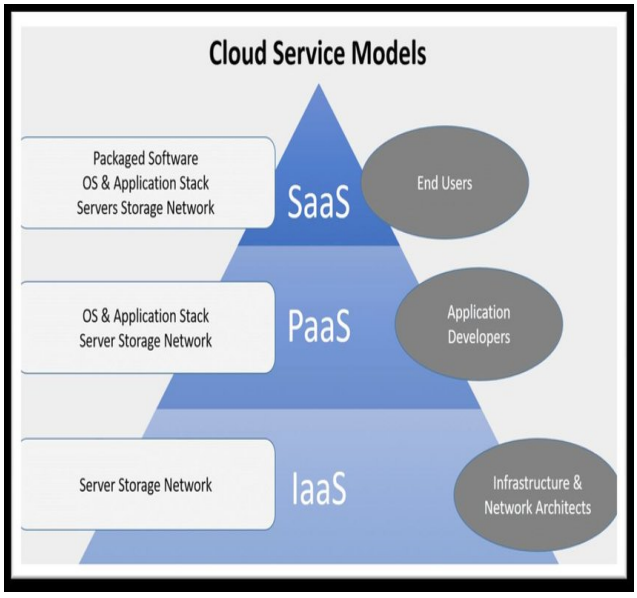

Figure 1: Cloud Service Delivery Model

3. Research Methodology

The researcher accessed different reference articles or reviews in the literature and cloud service providers' websites. Since this is an original research study, a new framework was developed in which services were delivered under the control and measurement of the two parties cloud services provider and the cloud services consumer. The problem was identified in a good manner. The researcher developed the new framework after the problem was analyzed. Finally, the implementation of the framework was done using a prototype and was evaluated according to the given criteria. This is illustrated as indicated in (figure 2).

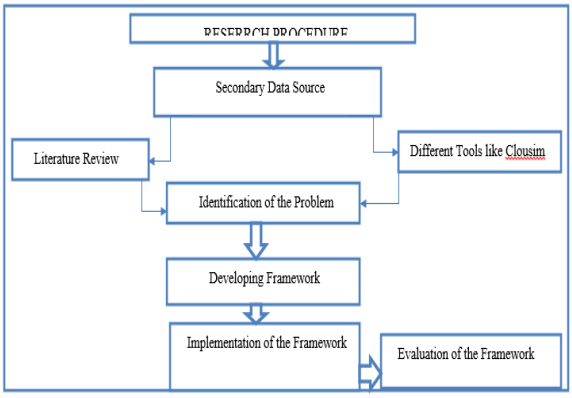

Figure 2: Flow of the Research Methodology Procedure

a. Tool Selection

The research was carried out using different tools like Microsoft Office for documentation, and Unified Modelling Languages (Figure 3). For the development of the prototype and simulation purposes, the researcher used Java programming languages and classism, respectively. The developed framework was evaluated based on the criteria. In this study, the MySQL database was used to store the delivered services under SLA that guided the primary cloud service actors [16].

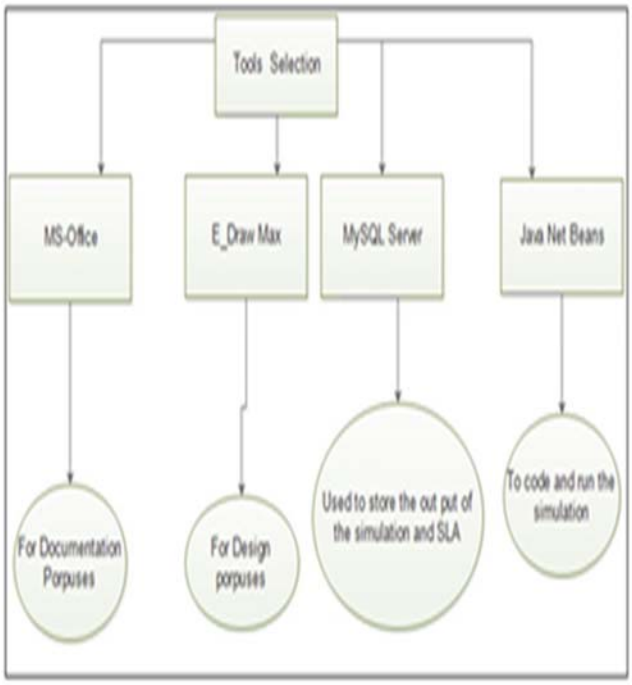

Figure 3: Tool used in the study for the indicated purposes

4. Framework Design & Results

The study was mainly focused on the solutions for the problems that occurred during service delivery which related to cloud-based service level agreements (figure 4). According to the investigation in this study, the majority of cloud providers monitor and measure cloud service level agreements exclusively for the sake of their own business, which causes big challenges for cloud service consumers. Remarkably, the findings were impressive and have the potential to provide the best solution to the current problem. If the service level agreement is violated, cloud providers hide the parameters that were under the quality of services mentioned in the service level agreement, making service consumers unhappy and causing insight. This service level agreement is controlled and monitored by both cloud providers and consumers, rather than just the cloud service provider [8,13,17].

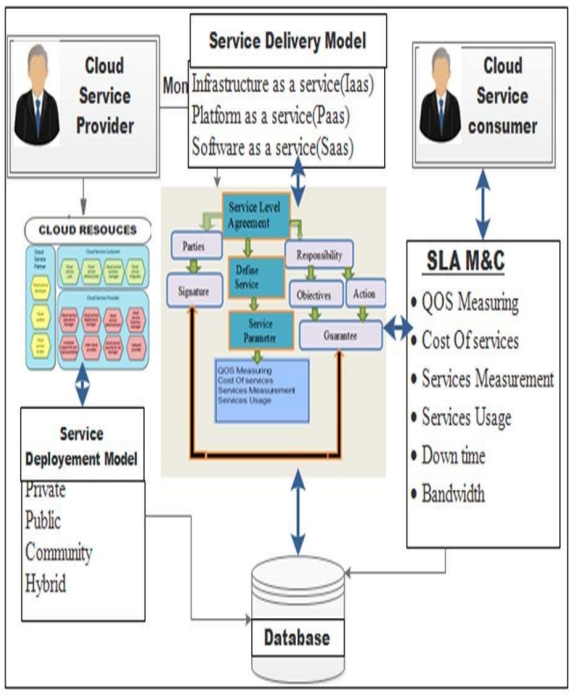

Figure 4: a developed framework for M&C SLA for both server and client-side

The developed framework was composed of different main components like cloud service provider, cloud service consumer, cloud service delivery, cloud resources, Service level agreement, service deployment models, and SLA M&C (Service level agreement monitoring and controlling). These main components consist of their components. The cloud service provider is responsible for providing services to the cloud service consumer under the service level agreement [18,19]. As a result, the cloud provider should control the cloud-based resources including service level agreement. The second component is the cloud service consumer, who should pay for the services they use on a pay-per-user principle. The service level agreement should be monitored and controlled by both the cloud service consumer and cloud service providers in this framework. Because there are various issues when the service level agreement is only monitored and controlled from the cloud service provider's side. This framework was the solution for the challenges like a mismatch of service delivered and what is mentioned under the service level agreement. This framework was designed to address this issue by including a basic component known as SLA M&C (service level agreement monitoring and control). The cloud service consumer can control and monitor the service level

agreement by reading the service level agreement document from the database/data center and comparing it to the services delivered. This mechanism works by examining service quality characteristics, such as cost of service, service measurement, service usage, downtime, and bandwidth.

5. Discussion

Different researchers have done their studies on cloud computing service level agreement over the last decades. In almost all cloud computing research related to service level agreements, the researchers were focused on cloud service provider side service level agreements. But monitoring service level agreement from the cloud service provider side does not answer the questions of cloud consumers. Cloud service consumers should know the principles of pay per use should not be compromised. Because of different factors cloud consumers are paid beyond the services delivered and as a result, the service level agreement was violated. When the service level agreement is violated most of the cloud providers hide the issue internally and the consumers had no information about how much the services are delivered. This is because the service level agreement was monitored only by cloud services providers. In this research, the problems were identified using different mechanisms or methods like secondary resources (literature review) and analyzed in I well manner. The study was carried out using different tools and java programming language to develop the prototype. The researcher used different criteria to validate the developed framework i.e. to validate SLA that is concerned with a cloud service provider, validate what happened when the request from the client-side is less than what is specified in SLA and above what is specified in SLA as well as implementing the monitoring mechanism using the developed Monitoring component. The researcher observed that with the 1st and 3rd criteria the service level agreement was violated and this indicated that if the Service level agreement is monitored or managed only by cloud service prover, there is a violation of LSA.

This implied that there are chances of the inconsistencies in resources provided to make the selves beneficiary beyond SLA under CSP and CSC was victimized while receiving services from CSP (figure 6). SLA M&C was used to compare the service level agreement signed between the two parties cloud service provider and cloud service consumer and the features of quality of services. This component makes alarms the cloud service consumers if the service level agreement is violated and the negotiation must be made with the cloud service provider to continue as a business partner.

Therefore, the researcher recommended that the service level agreement should be managed by both cloud service provider and service consumer over the cloud computing environment. As a result, the researcher evaluated the developed framework and approved as it a solution for the problems mentioned in the study.

.

Test Case	Cloudlet Parameters	Required /agreed Value	Allocated Value	Difference Value	Remark
	The first scenario which is the resource allocated are the same with the SLA Promised				
1	Length	50000	<50000	yes	violation
	File size	400	<400	yes	violation
	Output size	200	<200	yes	violation
	Core	1	1	No	No violation
	The scenario when request is less than what is mentioned under SLA				
2	Length	50000	< 50000	no	No Violated
	File size	400	<400	no	No Violated
	Output size	200	<200	no	No Violated
	Core	1	<1	no	No Violated
3	**The scenario when request is greater than what is mentioned under SLA**				
	Length	50000	>50000	yes	Violated
	File size	400	>400	yes	Violated
	Output size	200	>200	yes	Violated
	Core	1	>1	yes	Violated
4	**After an implementation of the developed framework All problems are solved no violation at all because the SLA_M&C was notify the CSC & CSP at the same time.**				
	Length	50000	=50000		No Violated
	File size	400	=400		No Violated
	Output size	200	=200		No Violated
	Core	1	=1		No Violated

Figure 5: The evaluation of all the scenarios

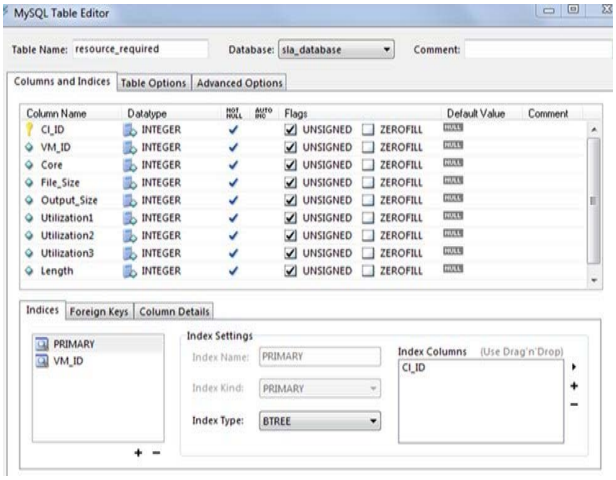

Figure 6: Shows SLA signed between CSC&CSP

6. Sample code for implementation

```
import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.LinkedList;
import java.util.List;
import org.cloudbus.cloudsim.Cloudlet;
import org.cloudbus.cloudsim.CloudletSchedulerTimeShared;
import org.cloudbus.cloudsim.Datacenter;
import org.cloudbus.cloudsim.DatacenterBroker;
import org.cloudbus.cloudsim.DatacenterCharacteristics;
import org.cloudbus.cloudsim.Host;
import org.cloudbus.cloudsim.Log;
import org.cloudbus.cloudsim.Pe;
import org.cloudbus.cloudsim.Storage;
import org.cloudbus.cloudsim.UtilizationModel;
import org.cloudbus.cloudsim.UtilizationModelFull;
import org.cloudbus.cloudsim.Vm;
import org.cloudbus.cloudsim.VmAllocationPolicySimple;
import org.cloudbus.cloudsim.VmSchedulerTimeShared;
import org.cloudbus.cloudsim.core.CloudSim;
import org.cloudbus.cloudsim.provisioners.BwProvisionerSimple;
import org.cloudbus.cloudsim.provisioners.PeProvisionerSimple;
import org.cloudbus.cloudsim.provisioners.RamProvisionerSimple;
// VM description
int vmid = 0;
int mips = 1000;
long size = 10000; // image size (MB)
int ram = 512; // vm memory (MB)
long bw = 1000;
int pesNumber = 1; // number of cpus
String vmm = "Xen"; // VMM name
// Cloudlet properties
int id = 0;
long length = 400000;
long fileSize = 300;
long outputSize = 300;
UtilizationModel utilizationModel = new UtilizationModelFull();
Cloudlet cloudlet = new Cloudlet(id, length, pesNumber, fileSize, outputSize,
utilizationModel, utilizationModel, utilizationModel);
cloudlet.setUserId(brokerId);
cloudlet.setVmId(vmid);
int hostId = 0;
int ram = 2048; // host memory (MB)
long storage = 1000000; // host storage
int bw = 10000;
hostList.add(
new host(s)
hostId,
new RamProvisionerSimple(ram),
new BwProvisionerSimple(bw),
storage,
peList,
new VmSchedulerTimeShared(peList)
)
private static void printCloudletList(List<Cloudlet> list) {
int size = list.size();
cloudlet cloudlet;
String indent = " ";
Log.printLine();
Log.printLine("========== OUTPUT ==========");
Log.printLine("Cloudlet ID" + indent + "STATUS" + indent
+ "Data center ID" + indent + "VM ID" + indent + "Time" + indent
+ "Start Time" + indent + "Finish Time");
DecimalFormat dft = new DecimalFormat("###.##");
for (int i = 0; i < size; i++) {
cloudlet = list.get(i);
Log.print(indent + cloudlet.getCloudletId() + indent + indent);
if (cloudlet.getCloudletStatus() == Cloudlet.SUCCESS)
Log.print("SUCCESS");
Log.printLine(indent + indent + cloudlet.getResourceId()
+ indent + indent + indent + cloudlet.getVmId()
+ indent + indent+ dft.format(cloudlet.getActualCPUTime()) + indent +
dft.format(cloudlet.getExecStartTime())                    + indent +
indent+ dft.format(cloudlet.getFinishTime()));
}
}
```

7. Conclusion

Cloud service providers have controlled systems or mentoring tools that ignore the calculation and do not take into account service defects. The main objective of the study was to remove the monopolization of controlling and monitoring cloud service level agreements by cloud service providers and to create the possibility of monitoring and controlling from both cloud service providers and cloud service consumers. So the researcher achieved what is specified as a projective during the study. During this study, the researcher used different tools and programming languages, including database servers. This framework component is used to monitor and control by comparing the services stated in the service level agreement and the services delivered to the cloud service consumer using different measurements, as demonstrated in a prototype implementation. Both the cloud service provider and cloud service consumers can monitor the delivered service to the consumer by considering the defects such as cost, downtime, service delay, and overloading. The cloud service consumer can control and monitor the service level agreement by comparing delivered services including quality of service, availability, and so on with the stated agreement under the service level agreement. As a result, this framework created a new knowledge base to provide consumers with transparency in monitoring and controlling cloud-based service level agreements. As a result, the researcher evaluated the developed framework and approved as it a solution for the problems mentioned in the study.

Conflicts of Interest

The authors declare no conflict of interest, financial or otherwise.

Availability of Data and Materials

All data generated or analyzed during this study were included in the parent document, therefore, no additional data were available.

Funding

The author declares that no financial grant was used for this study.

Acknowledgments

The author sincerely acknowledged his wife Seble Seyoum and his daughter Fenan Abel.

References

1. Velte, A.T.; Velte, T.J.; Elsenpeter, R.C.; Elsenpeter, R.C. Cloud computing: a practical approach. **2010**.
2. Mell, P.; Grant, T. The NIST definition of cloud computing. **2011**.
3. Alade, A.J.; Sharma, R.K. Convergence of Intranet in project management for Effective Enterprise Management. **2008**.
4. Dharmveer Yadav, D. Study of Intranet over Cloud. *International Journal of Innovation in Engineering and Technology* **2017**, *16*.
5. Sharma, D.P.; Sharma, A.K.; Shekhawat, N.S. The best performance practices in project management of SAP ERP accomplishment. *International Journal of Information and Electronics Engineering* **2012**, *2*, 835-839.
6. Shekhawat, H.S.; Sharma, D.P. Hybrid cloud computing in E-governance: Related security risks and solutions. *Research Journal of Information Technology* **2012**, *4*, 1-6.
7. Shekhawat, N.S.; Sharma, D.P. Cloud Computing Security through Cryptography for Banking Sector. *Computing for National Development, New Delhi* **2011**.
8. Patel, P.; Ranabahu, A.H.; Sheth, A.P. Service level agreement in cloud computing. **2009**.
9. Authority, A.P.R. Outsourcing involving cloud computing services. **2018**.
10. Choi, T.; Kodirov, N.; Lee, T.-H.; Kim, D.; Lee, J. Autonomic management framework for cloud-based virtual networks. 2011; pp. 1-7.
11. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.H.; Konwinski, A.; Lee, G.; Patterson, D.A.; Rabkin, A.; Stoica, I. *Above the clouds: A Berkeley view of cloud computing*; Technical Report UCB/EECS-2009-28, EECS Department, University of California …: 2009.
12. Adane, A. Cloud Services Quality Enhancement through Control & Monitoring of Service Level Agreements at Server and Consumer Side. *Journal of Computer and Communications* **2020**, *8*, 78.
13. Therese, L.; Rogers, B.C. An effective Trust based access control service for adaptive monitoring cloud computing. *Networking and Communication Engineering* **2015**, *7*, 415-419.
14. El-Awadi, R.; Abu-Rizka, M. A framework for negotiating service level agreement of cloud-based services. *Procedia Computer Science* **2015**, *65*, 940-949.
15. Gebreslassie, Y.; Prasad Sharma, D. DPS-Yemane-Shareme CSMM Model for Client-Side SLA of Green Cloud Services Measuring and Monitoring. *IUP Journal of Computer Sciences, July* **2019**.
16. Abel, A. An Effective Framework To Monitor Service Level Agreements In Cloud Computing Environment. **2017**.
17. Ludwig, H.; Keller, A.; Dan, A.; King, R.P.; Franck, R. Web service level agreement (WSLA) language specification. *IBM corporation* **2003**, 815-824.
18. Emeakaroha, V.C.; Brandic, I.; Maurer, M.; Dustdar, S. Low-level metrics to high-level SLAs-LoM2HiS framework: Bridging the gap between monitored metrics and SLA parameters in cloud environments. 2010; pp. 48-54.
19. Wu, L.; Buyya, R. Service level agreement (SLA) in utility computing systems. In *Performance and dependability in service computing: Concepts, techniques and research directions*; IGI Global: 2012; pp. 1-25.

8.

Abel Adane was born in Chancho, Oromiaya, Ethiopia in 1989. He received the B.S. and M.S. degrees in Information Technology and Software Engineering from the Debre Birhan and Adama Science and technology University, Ethiopia Respectively. From 2013 to 2015, he was a Graduate Assistant with computer Science department, Arba Minch University. His research interests include cloud computing and software engineering filed of areas.