© 2018 IJSRSET | Volume 4 | Issue 2 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
National Conference on Advanced Research Trends in Information and Technologies (NCARTICT-2018), Department of IT, L. D. College of Engineering, Ahmedabad, Gujarat, India In association with
International Journal of Scientific Research in Science, Engineering and Technology

Service Level Agreements & its Impact on Cloud

Pranav Tank, Darshan Upadhyay, Kunal Khimani
Asst. Professor, I.T. Department, VVP Engineering College, Rajkot, Gujarat, India

ABSTRACT

With the growth of technology, there has been scalable increase in use of Cloud services in real world applications. Cloud becomes efficient as it works on Pay-as-you-use basis. So with increasing number of cloud service, Service Level Agreement between Service Provider (seller) and Service Consumer (customer) becomes more important which will guarantee service type and its quality. Not just withstanding the point of quality of service, SLA also works on monitoring the service and refining the parameters which can increase the level of service. The objective of this paper is to present comprehensive study about how SLA works in a cloud environment and how it can impact in real world applications.

Keywords: Cloud, Pay-as-you-use, SLA, Quality.

I. INTRODUCTION

Cloud computing is term used in numerous definitions by many research scholars. But in general terms it is defined for delivery of hosted service over the internet. Cloud means internet and computing means type of internet-based computing, where services such as storage, servers, applications, etc are delivered to consumer through internet. Basically usage of Cloud has been increased due to pay-as-you-use service implemented in applications. Service level agreement is more unified version established through negotiation between service consumer and provider.

1.1 Types of Cloud

Cloud is specified in mainly 4 categories: Public, Private, Hybrid and Community. The Community Cloud is rarely used and so in this paper we will be over viewing previous three versions only.

1.1.1 Public Cloud

Public Cloud is implemented for general use. Data remains publicly available to all users. Users are charged for the time duration they use the services, like usage of CPU cycles, bandwidth or storage consumed, etc [1]. Usually, a third party cloud service delivers cloud service over the internet. They are more vulnerable to security threats than other cloud models. Example of Public Clouds is Amazon Web Service (AWS), Microsoft Azure, IBM Websphere, etc.

Figure 1. Public Cloud

1.1.2 Private Cloud

Private Cloud services are delivered between business data center to internal users. Users may not be charged for the services. This is more secure and controlled model of cloud. Example of Private Clouds includes VMware, OpenStack, etc.

Figure 2. Private Cloud

1.1.3 Hybrid Cloud

Hybrid Cloud services is combination of Public cloud and on-premises Private Cloud. Goal of Hybrid cloud is to use infrastructure of Public Cloud and still maintain vulnerability of the data.

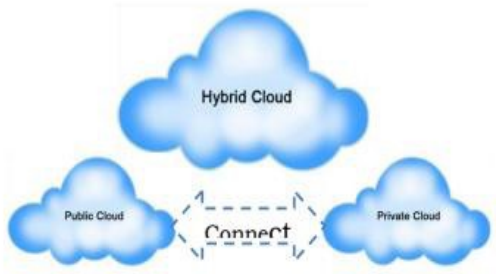

Figure 3. Hybrid Cloud

1.2 Characteristics of Cloud

On-Demand Service: These are services required by the customer to manage their resources. Service is provided over internet by provider to customer so that they can manage their own computing resources.

Resource Pooling: Cloud service provides pool of resources that can rapidly provisioned and be released with minimal efforts. Consumers can access resource directly from remote data centre.

Elasticity: Cloud services can be requested or managed from cloud providers as per customers request. Elasticity means service can be scaled-up or scaled-down as per customer requirement.

Measured Service: Services are billed according to customers requirements. Customers demand defines services required.

1.3 Service Models of Cloud

Cloud computing is computational process in which service are delivered over network using computing resources. There are three main types of service models:

- ✓ Software as a Service (SaaS)
- ✓ Platform as a Service (PaaS)
- ✓ Infrastructure as a Service (IaaS)

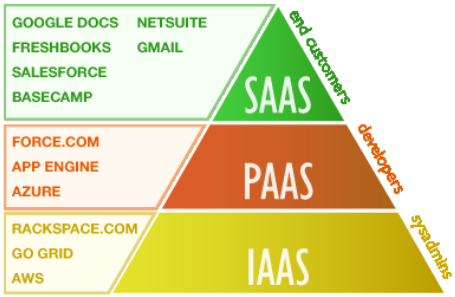

Figure 4. Cloud Service Models

1.3.1 Software as a Service (SaaS)

In this service model, consumer uses application running on a cloud infrastructure. Few popular examples for SaaS, as shown above in Fig. are Salesforce, Netsuite, Basecamp etc.

1.3.2 Platform as a Service (PaaS)

In this service model consumer uses platform which includes all system and environments consisting all phase of SDLC life cycle. Few popular examples of PaaS, as shown in Fig. are Azure, App Engine etc.

1.3.3 Infrastructure as a Service (IaaS)

In this service model provider delivers the user the infrastructure over the internet. User is able to deploy and run various softwares including application or system softwares. Few popular examples of IaaS, as shown in Figure are Rackspace.com, AWS etc.

II. GENERAL OVERVIEW OF SERVICE LEVEL AGREEMENT (SLA)

Main components of SLA include SLA Definition, SLA components, Customer and Service Provider Definition and Qualities & Parameters determined by SLA.

2.1 SLA Definition

"Service Level Agreement" is defined as format that contains explanation of the agreed service, parameters of the level of service, the guarantees regarding the Quality of Service, and arrangements for all cases of violations. In short, SLA is the tool for protecting the stability of the service[3]. The SLA is very significant as a contract that is held between the provider of the service and another party who could be one of following; consumer of the service, broker negotiator, or monitoring negotiator. SLA can be considered as quantifiable contract, the service that service provider will provide

and what sanction service provider will pay if the dedicated objective is not met.

Parameters that SLA includes for Cloud service are Web services, Networking, Internet, Data centre management, etc. Most summarized SLA contains terms like: business endings, pricing strategy etc.

2.2 SLA Components

The more refined SLA improves customer acceptance level, as it supports provider to focus and work on needs prioritized by customers. It also helps to check whether process is on the right path. Consumer can even check the quality and examine the service based on Service Level Objective (SLO) determined in SLA.

Quality of Service (QoS) is one of the Enhanced terms which is measured with Key Performance Indicator (KPI) component in SLA. Customers check whether in the agreed contract whether or not these indicators match Service Level Objective (SLO). SLO contains levels of parameters, specific value, behaviour of services etc as a target to achieve. Actual parameter values are compared with stated ones to evaluate the performance. These indicators are used to check and test these parameters and are used later in determining the violation.

2.3 Consumer and Provider definition

The word Service Provider refers to firm or business which provides service. Service supplier may be network, worker, transporter, Internet Service Provider (ISP) or an Application Service Provider (ASP)[4]. The word Service Customer refers to firm or organization that employs the service supplied by the service provider. Service providers exploit this base to enhance their usage of infrastructure to achieve signed conditions of services. Service consumers exploit the SLA to reach the stage of quality of service they require and to keep appropriate business models for a long period.

2.4 Parameters determined by SLA

There are two groups of qualities that are identified by SLA: measurable and un-measurable qualities. Measurable qualities could be measured in metrics; whereas Un-measured qualities could be measured from specified estimation. Metrics are exploited in monitoring procedure, software procedure enhancement, business policy employment, and mainly any area where information has to be gathered to confirm whether objectives are being achieved. These metrics decides the amounts needs to be collected to confirm whether SLA parameters are achieved[6]. Note that, if the parameters are declared and negotiated, the changes needs to be applied to both consumer and provider side[5]. Following are the parameters:

- Reasonable, which allows contributor to encourage to perform well. For example, SLA parameter identifies cost depending on amount of service used.
- Achievable, all the parameters needs to be included in the metrics. For example, unexpected kind of communication over the internet.
- Quantifiable, metrics must be measurable and permit for measurements. For example, if gathering metrics exploits important resources, it might not deserve to spend any effort.

III. LIFE CYCLE OF SLA

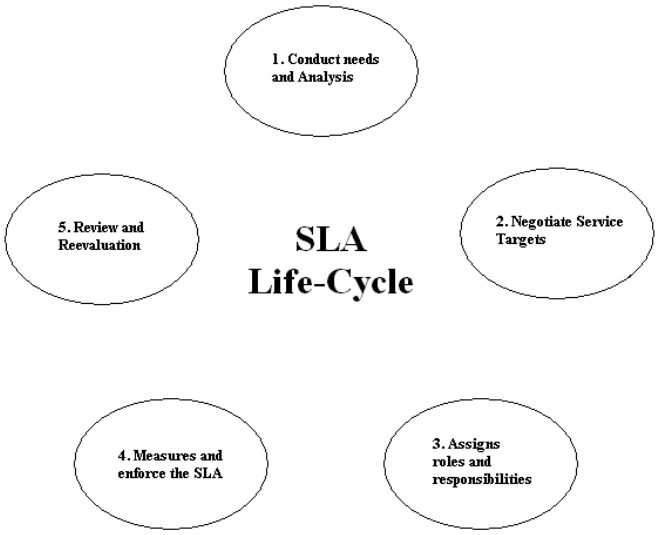

Figure 5. SLA Life Cycle

SLA includes 5 stages to be completed. These stages consists of; Needs and Analysis, Negotiation, Assigning roles and responsibilities, Measures and enforcement and Review[3].

3.1 Needs and Analysis:

This stage includes the identification of customer requirements and needs, the network capabilities, the identification of the suitable service features and parameters, service's levels, service executional environment, and the implementation of the standard of SLA templates.

3.2 Negotiate Services:

This stage includes the negotiation of an SLA with the consumer to select the values of SLA parameters related to specific services, the costs gained from the service customer after signing the SLA, the costs incurred by the service provider when the SLA is violated, the definition and at last periodicity of the reports associated with service to be delivered to the service customer.

3.3 Assign Roles and Responsibilities:

This stage include the service's resource provisioning, where the service is enabled and prepared for the consumption, configuration of the network which might be to achieve specific requirements in the service, or to support the service network overall, and service activation. Service provisioning and deployment stage may need the reconfiguration of the service resources to support the executional stage which will lead to a successful achievement of the SLA parameters. Thus it becomes very important to assign roles of each user and make them aware of the strategy they need to work on.

3.4 Measure and enforce SLA:

This stage includes measurement and assessment of the SLA. Assessment is done in two phase, assessment with the individual customer and overall service assessment. In first phase reviewing QoS, possible enhancement and altering requirements are examined. Whereas, in second phase, readjustment of goals, support problems and establishing service levels are measured. At this phase we would propose our work.

3.5 Review and Evaluation of SLA:

Once the SLA is signed, however, if job isn't over; SLA has to enforce the guidelines both (consumer + provider) agreed to. To evaluate the SLA one should also include Master Service Agreement (MSA) to compare each parameter. MSA defines services and liabilities in general terms, which will be same for all the consumers. Moreover, SLA is specific for the client and will do exactly that is expected.

IV. PROPOSED WORK

Generally the work proposed till now includes changes in SLA parameters, reducing the factors affecting violation of SLA or maintaining/upgrading the threshold value of SLA. But in our work, we would like to propose an automation system which focuses on MSA rather than on SLA. To upgrade the parameters, majorly parameters included in SLA are compared with MSA. So before comparison we can develop a automation system which compares the parameters predefined in MSA with the parameters included in SLA. The changes can be marked directly between them. If the difference is large the system will upgrade the required parameter/s directly so as to build a more effective SLA. This will also help in reducing the violation ratio.

Following are the parameters which are predefined in MSA and also required while designing a SLA.

Table 1. Parameters included for designing SLA in Cloud Environment.

Object	No. Of metrics Parameters	General Measurement (Unit)
Hardware	11	Time/Number
Software	05	Time
Network	10	Time/Percent
Storage	10	Time/Percent/Number
Service Desk	09	Percent/No. ff Language

4.1 Working of Metric Parameters

As shown in Table-1, Hardware includes parameters like Availability, Response Time, Failure Frequency, Processor Time, and Instruction per second, Number of Workstation etc. Software includes parameters like Service Time, Solution Time, and Number of Licenses etc. Network includes, Availability LAN, Availability WAN, Access RAS, Latency times, etc. Storage includes, Max. Down Time, Backup, Memory Size, Periods of Operation, etc and Service desk includes parameters like, Failure forwarding degree, Language Variety, Self Solution rate etc.

These objects and metrics are already included in Master Service Agreement (MSA). During SLA design, which differs from consumer to consumer depending on requirement, these parameters are taken into consideration. Thus, a well-versed SLA is formed, which will provide required QoS. Ultimately we look for better SLA which provides best Quality of Service. A way of increasing QoS is by changing the Key Performance Indicator (KPI) value of the parameter. But this will not be that effective as it points to the same

problem of changing the parameters after/on SLA violation.

Existing system compares the parameters only after/on SLA violation. So far seen system works on Metrics and Non-metrics parameters of SLA. While in the proposed system parameters are included depending on the customer's requirement. Depending on this requirement Service Level Objective (SLO) responsible for designing a SLA, decides whether to include the parameter or to upgrade the parameter. If the Service Level Objective is achieved above/just at the threshold value, parameter is included in SLA. If the Service Level Objective is achieved below the threshold value, parameter is considered for update and rejected.

So, we focus on parameters included in High-Level Metrics and Low-Level Metrics, for designing a SLA. Therefore, parameters that are required for designing are included in SLA using SLO mechanism. If required, parameters are upgraded/changed first and are included, this reduces the probability of violation of SLA to approx 0%. So it becomes easy for the system to compare/review the threshold for designing better SLA. This system will be beneficial for not only preventing SLA violation but also in achieving required QoS pre-defined in SLA.

V. CONCLUSION AND FUTURE WORK

Our studies on number of SLAs currently used throughout have revealed that today's prevailing contract focuses on QoS metrics, namely Availability, Response Time etc. Other parameters are mostly never mentioned and so the steps are taken for reducing the violation of SLA.

This proposed work will focus on the basics of SLA and not on reducing the violation of SLA but preventing the violation of SLA. The collection of Metrics, both high-level and low-level, will assist for developing this automation system. These metrics are easy to automate and commonly used in IT process and service. Categorization scheme is populated by updating categories, metrics, addition of parameters, etc. In summary, the proposed categories are useful to find an initial problem and possible metrics which can be automated and used as measure of performance. With the varying taxonomical structure of an enterprise, level of categories / subcategories tends to change in future.

VI. REFERENCES

[1]. Pranali Gajjar, Brona Shah: An Efficient Scalable Framework for Auto Scaling Services in Cloud Computing Environment, IJIRT, Volume 2 Issue 12, May 2016.

[2]. Hayes, I.: Five Principles for Selecting SLA Metrics, http://www.clarityconsulting.

[3]. com/five_principles.htm, Clarity Consulting, 2004.

[4]. Paschke, A., M. Bichler, Jens Dietrich ContractLog: An Approach to Rule Based

[5]. Monitoring and Execution of Service Level Agreements. Int. Con. on Rules and Rule

[6]. Markup Languages for the Semantic Web (RuleML 2005), Galway, Ireland, 2005.

[7]. Lijun Mei, W.K. Chan and T.H. Tse, "A Tale of Clouds: Paradigm Comparisons and Some Thoughts on Research Issues", IEEE Asia-Pacific Services Computing Conference, 2008, pp 464-469.

[8]. R. Pragaladan, P. Suganthi, "A Study on Challenges of Cloud Computing in Enterprise Perspective", July 2004.

[9]. M.Kriushanth, L. Arockiam, and G. Justy Mirobi, "Auto Scaling in Cloud Computing: An Overview", July 2013.