SLAs in Cloud Systems: The Business Perspective

[1]Rabi Prasad Padhy, [2]Dr. Manas Ranjan Patra, [3]Dr. Suresh Chandra Satapathy
[1]Oracle Corporation, Bangalore, Karnataka, India
[2]Dept. of Computer Science, Berhampur University, Berhampur, Odisha, India
[3]Dept. of CSE, ANITS, Sanivasala, Andhra Pradesh, India

Abstract

Cloud computing is a pay-per-use model which enables service providers to offer their services to customers in different Quality-of-Service (QoS) levels. The level of customer satisfaction is crucial in cloud computing and its success totally depends upon both customers and service providers can be confident that signed SLA's are supporting their respective business activities to their best extent. However, there are no metrics can be referred to compare these providers, so it is difficult for cloud consumers to select the most reliable providers or resources. A main challenge for a service provider is to manage (Service Level Agreements) SLAs for its service consumers, i.e. automatically determine the appropriate resources required from the lower layer in order to respect the QoS requirements of his consumers. Service Level Agreements (SLAs) are negotiated agreements of common understanding, between customers and service providers, which define the services provided along with their expected levels of performance. They help manage the strategic relationship between companies and Outsourcing service providers, hence the importance of SLA management has increased with the growth of Outsourcing in recent years. In this paper, we motivate the need for a SLA in cloud computing and a comparative study of different cloud providers SLA's. This paper also addressed SLA negotiation, mapping and management in context of cloud service selection and adoption.

Keywords

Cloud Services, SLA's, Quality of Service, SLA Negotiation, SLA Management, SLA Mapping, SLA Enforcement

I. Introduction

Service Level Agreements (SLAs) are a common way to formally specify the exact conditions (both functional and nonfunctional) under which services are or are to be delivered. However, SLAs in practice are specified at the top-level interface between a service provider and a service customer only. Customers and providers can use top-level SLAs to monitor whether the actual service delivery complies with the agreed SLA terms. In case of SLA violations, penalties or compensations can be directly derived [1]. Traditionally, service level agreements (SLAs) have been used as instruments to formalize the roles of parties to a contract and to clearly state expectations, penalties for violations and other contractual terms (E.g. legal and pricing). In the service economy (Grid computing, clustering, P2P, Cloud Computing etc), the need to automate SLAs has been well motivated due to cost-effectiveness, transparency, accountability and improved monitoring of provisioned services,. Patterns and metric measurements from such SLAs could be used as input into more sophisticated mechanisms for resource management and price optimization.

The emergence of Cloud computing solutions has attracted many potential consumers, such as enterprises, looking for a way to reduce the costs associated with supporting their business processes. Using the Cloud, these customers can outsource services (offered by service providers) which can be easily composed to build distributed applications [2]. Additionally, the Cloud allows resource providers to offer raw resources as a service where the consumers can outsource the execution of their own services. In the same way, service providers can also use the Cloud to overcome an unexpected demand on their services by outsourcing additional resources to other providers, or even they can fully rely on external resources to provide their services [3]. In all these scenarios, it is highly desirable that the consumers receive fine-grain Quality of Service (QoS) guarantees from the providers. However, this becomes essential when service providers outsource resources to resource providers, since the QoS guarantees they can provide to their respective customers depend on the QoS guarantees they receive from the resource providers. Typically, a provider agrees the QoS with its customers through a Service Level Agreement (SLA), which is a bilateral contract between the customer and the provider (or between providers) that states not only the conditions of service, but also characterizes the agreed QoS between them using a set of metrics. Service providers naturally offer service-level metrics (e.g. service execution deadline) to their customers for specifying the QoS. Using service-level metrics has advantages for both the customer and the provider [4]. The former does not need to provide detailed information about the resource requirements of the service to be executed (probably the customer does not know this exactly), but only a high-level performance goal (e.g. a deadline for executing the service). The latter can freely decide the allocated resources to the service whereas it guarantees that the service meets the agreed performance goals. Being able to dynamically allocate resources to the different services is especially important for Cloud providers considering that, aiming for profitability, they tend to share their resources among multiple concurrent services owned by different customers. Service Level Agreements are established between service consumers and providers and define a number of obligations and rights for both sides [5]. Research on SLA management for web services is often focusing on SLAs with rights for consumers and obligations for providers, keeping the balance by requiring per-use or interval payment by the consumer. To attract consumers, providers need to clearly specify the pricing of service offerings. SLAs are defined prior to service usage based on a negotiation between both sides, which can happen autonomously within any number of constraints, or can be performed interactively, even offline in some cases [6]. Online negotiation facilities based on agents and SLA managers are increasingly available and support more dynamic service environments. In most negotiation protocols, SLA templates or profiles are offered by the service provider to initiate the negotiation. The providers either represented by humans or by an SLA manager service, need to know in advance of how to find a suitable ratio of payment to operational cost in order to create feasible SLA templates. This is a non-trivial task in the context of distributable services whose functional and especially non-functional properties cannot be inferred from the associated service descriptions or SLA templates [7]. The purpose of using SLAs is to define a formal basis for performance and availability the provider guarantees to deliver. SLA contracts record the level of service, specified by several attributes such as availability, serviceability, performance, operation, billing or even penalties in the case of violation of the SLA. Also, a number of performance-related metrics are frequently used by Internet Service Providers (ISPs), such as service response time, data transfer rate, round-trip time, packet loss ratio and delay variance.

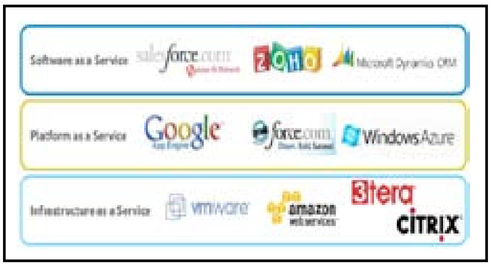


Fig. 1: Cloud Computing Business Model

II. Cloud Service Level Agreement

A. What is an SLA?

An Service Level Agreement is a document that describes the minimum performance criteria a provider promises to meet while delivering a service. It typically also sets out the remedial action and any penalties that will take effect if performance falls below the promised standard. It is an essential component of the legal contract between a service consumer and the provider.

B. What is a Cloud SLA?

An Agreement between an Cloud Service Provider and a Customer. The SLA describes the Cloud Service, documents Service Level Targets, and specifies the responsibilities of the Cloud Service Provider and the Customer.

C. What is an Service Level Objectives (SLOs)?

An SLA contains service level objectives (SLOs) that define objectively measurable conditions for the service; some examples include parameters of throughput and data streaming frequency and timing, availability percentages for VMs and other resources and instances, or urgency ratings to rank the importance of different SLOs (like "availability is more important than response time"). SLO expectations should vary depending on whether applications and data the applications access are hosted on the same cloud or on different ones [8].

D. Key Elements of a Cloud Service Level Agreement

- Server Uptime
- Persistent Storage
- Network Performance
- Load Balancing
- Cloud Storage
- Location of the data
- Access to the data
- Portability of the data
- Server Reboot
- Support Response Time
- Domain Name Services
- Security / privacy of the data
- Process to identify problems and resolution expectations
- Change Management process
- Dispute mediation process
- 7 x 24 x 365 Supports
- Service Termination

E. Typical Service Level Agreement Contents

1. Definition of Services

This is the most critical section of the Agreement as it describes the services and the manner in which those services are to be delivered. Standard services are often separated from customized services but this distinction is not critical. The information on the services must be accurate and contain detailed specifications of exactly what is being delivered [9].

2. Performance Management

A key part of a Service Level Agreement deals with monitoring and measuring service level performance. Essentially, every service must be capable of being measured and the results analyzed and reported. The benchmarks, targets and metrics to be utilized must be specified in the agreement itself. The service performance level must be reviewed regularly by the two parties.

3. Problem Management

The purpose of problem management is to minimize the adverse impact of incidents and problems. This usually specifies that there must be an adequate process to handle and resolve unplanned incidents and that there must also be preventative activity to reduce occurrence of unplanned incidents.

4. Customer Duties and Responsibilities

It is important for the customer to understand that it also has responsibilities to support the service delivery process. The SLA defines the relationship which of course is a two way entity. Typically, the customer must arrange for access, facilities and resources for the supplier's employees who need to work on-site [10].

5. Warranties and Remedies

This section of the SLA typically covers the following:

(i). Security

Security is a particularly critical feature of any SLA. The customer must provide controlled physical and logical access to its premises and information. Equally, the supplier must respect and comply with the Client's security policies and procedures.

(ii). Disaster Recovery and Business Continuity

Disaster recovery and business continuity can be of critical importance. This fact should be reflected within the SLA. The topic is disaster recovery is usually embraced within the security section. However, it is also frequently included within the Problem Management area. At the highest level, both these areas typically state that there must be adequate provision for disaster recovery and business continuity planning to protect the continuity of the services being delivered [11].

(iii). Termination

This section of the SLA agreement typically covers the following key topics:

- Termination at end of initial term
- Termination for convenience
- Termination for cause
- Payments on termination

6. Cloud SLA Definitions

(i). Downtime

means, for a domain, if there is more than a five percent user error rate. Downtime is measured based on server side error rate.

(ii). Downtime Period

Means, for a domain, a period of ten consecutive minutes of Downtime. Intermittent Downtime for a period of less than ten minutes will not be counted towards any Downtime Periods.

(iii). Monthly Uptime Percentage

means total number of minutes in a calendar month minus the number of minutes of Downtime suffered from all Downtime Periods in a calendar month, divided by the total number of minutes in a calendar month.

(iv). Scheduled Downtime

means those times where Google notified Customer of periods of Downtime at least five days prior to the commencement of such Downtime. There will be no more than twelve hours of Scheduled Downtime per calendar year. Scheduled Downtime is not considered Downtime for purposes of this Google Apps SLA, and will not be counted towards any Downtime Periods [12].

(v). Business Level Objectives

An organization must define why it will use the cloud services before it can define exactly what services it will use. This part is more organizational politics than technical issues: Some groups may get funding cuts or lose control of their infrastructure.

(vi). Responsibilities of Both Parties

It is important to define the balance of responsibilities between the provider and consumer. For example, the provider will be responsible for the Software-as-a-Service aspects, but the consumer may be mostly responsible for his VM that contains licensed software and works with sensitive data. Business

(vii). Continuity/Disaster Recovery

The consumer should ensure the provider maintains adequate disaster protection. Two examples come to mind: Storing valuable data on the cloud as backup and cloud bursting (switchover when in-house data centers are unable to handle processing loads).

(viii). Redundancy

Consider how redundant your provider's systems are [13].

(ix). Maintenance

One of the nicest aspects of using a cloud is that the provider handles the maintenance. But consumers should know, when providers will do maintenance tasks:

- Will services be unavailable during that time?
- Will services be available, but with much lower throughput?
- Will the consumer have a chance to test their applications against the updated service?

(x). Data Location

There are regulations that certain types of data can only be stored in certain physical locations. Providers can respond to those requirements with a guarantee that a consumer's data will be stored in certain locations only and the ability to audit that situation [14].

(xi). Data Seizure

If law enforcement seizes a provider's equipment to capture the data and applications belonging to a particular consumer, that seizure is likely to affect other consumers that use the same provider. Consider a third party to provide additional backup [15].

(xii). Provider Failure

Make contingency plans that take into account the financial health of the provider.

(xiii). Jurisdiction

Again, understand the local laws that apply to the service provider as well as consumer.

(xiv). Brokers and Resellers

If the service provider is a broker or reseller of cloud services, then consumer needs to understand the policies of the provider and the actual provider [16].

(xv). Client Agent

Responsible for collecting users' application requirements, for creating and updating the SLAs in order to grant always to best QoS [17].

(xvi). Negotiator

Manages SLAs and mediates between the user and the meta-broker, selects appropriate protocols for agreements, negotiates SLA creation, handles fulfillment and violation.

(xvii). Mediator

Select a vendor agent that is capable of deploying a service with the specified user requirements.

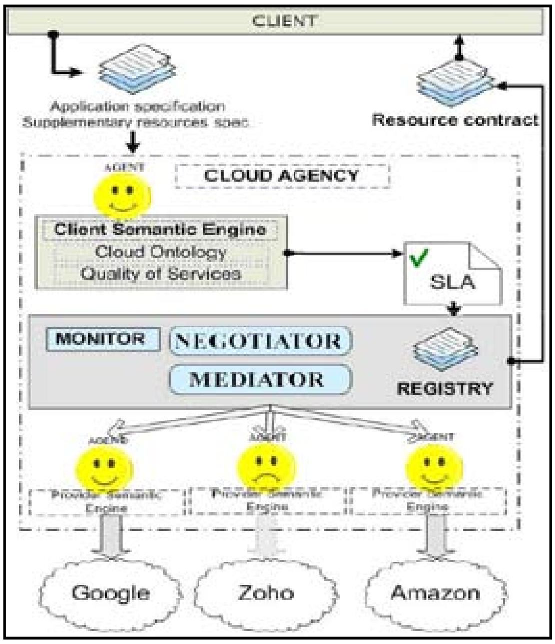


Fig. 2: SLA Based Cloud System Infrastructure

(xviii). Vendor Agent

Interacts with virtual or physical resources at provider side, and in case the required service needs to be deployed it interacts directly with the automatic service deployer [18].

(xix). Archiver

Stores historical data about quality of services and resources offered by providers

(xx). Service Deployer

Install the required service on the selected resource on demand

(xxi). Benchmarker

Periodically build performance figures of used resources and notify the client agents about values of measured parameters [19].

III. SLA Template

Due to the large variety in computing resources and, consequently, the large number of different types of Service Level Agreements (SLAs), any market for computing resources faces the potential problem of a low market liquidity. To counteract this problem, offering a set of standardized computing resources is appropriate. Each of these standardized computing resources is defined through an SLA template. An SLA template defines the structure of an SLA, the attributes, the names of the attributes, and the attribute values. Since these SLA templates are currently static, they cannot reflect changes in users' needs [20].

SLA Templates are the key component of SLA Negotiation and of Service Provider discovery, as templates are the tools that support the advertisement of services to the outside world and negotiation always begins with a client requesting an SLA Template from a provider [21].

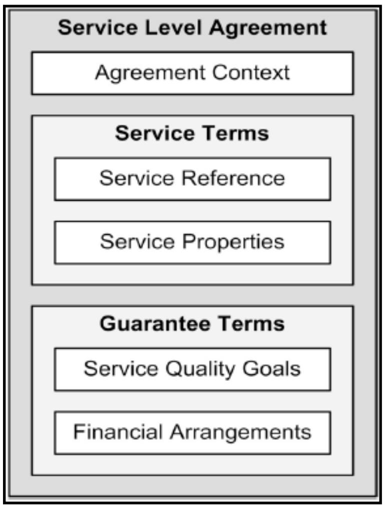


Fig. 3: Cloud SLA Template

IV. Cloud SLA Life Cycle

A SLA life cycle includes the following states: SLA template development, negotiation, creation, execution, monitoring of agreement states, and termination. Figure shows a typical SLA life cycle.

- Development of service and service templates,
- Discovery and negotiation of an SLA,
- Service provisioning and deployment,
- Execution of the service,
- Assessment and corrective actions during execution
- Termination and decommission of the service.

The semantics of the states is as follows:

In the SLA template development state, SLA templates are developed. Negotiation between service provider and consumer takes place in the negotiation state. SLA is created in the creation and executed in the execution state [22]. In the monitoring state, a monitoring system checks its services and guarantees. SLA is in the the termination state if the agreement terminates its life cycle. When an agreement is implemented, it does not imply that it is monitored. It remains not ready until the services covered by the agreement start their execution. Run-time renegotiating of agreements requires adding a new state, the renegotiation state, when renegotiation between service provider and consumer takes place. Run-time renegotiating occurs in case of a recoverable violation of an agreement term or in case the monitoring system is anticipating a possible violation of a term. Fig. shows an extended SLA life cycle [23]. An agreement is in the negotiation state during the negotiation process. From the negotiation state, the agreement can go to the creation state if the agreement is accepted by all the parties or to come abruptly to an end if it is rejected. An agreement is in the monitoring state while a monitoring system checks its services and guarantee. From the monitoring state, it is possible to go to five different states: to termination if the agreement finishes its life cycle; to denying if the agreement is violated and no negotiation terms can be applied, the agreement must terminate; to warning if the monitoring system issues at least one warning for at least one term; back to execution if the agreement is fulfilled; to renegotiated if the agreement is fulfilled or violated and negotiation terms can be applied [24].

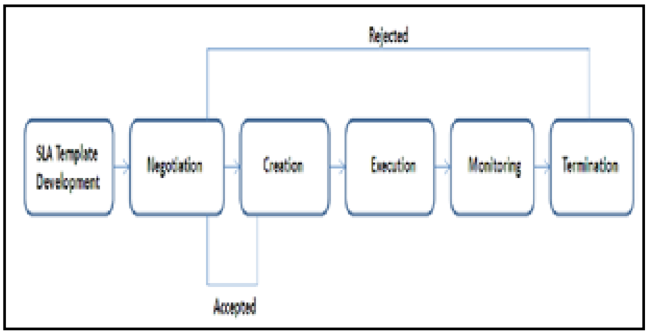


Fig. 4: Cloud SLA Life Cycle

A. SLA Negotiation Process

One of the preliminary requirements, which are relevant to support negotiation activities into the Cloud, is the definition of QoS parameters for existing service.

QoS parameters are necessary to:

- Fill the services request in order to negotiate the Cloud
- Resource
- Describe the Cloud offer
- Match the compliant services and build the best available solution
- Define the SLA
- Monitor the service levels

QoS parameter will be part of a common Cloud ontology to support interoperability and advanced semantic services. SLA negotiation with multiple Cloud providers will be delegated to a broker in a market based context that provide for:

- Searching for available Cloud services, compliant with user needs;
- Checking trustiness of providers;
- Deciding with whom to negotiate, according to user
- Requirements and past experiences;
- Negotiating the best price for the same offer by different providers;
- Negotiating of multiple SLAs, with different providers, to overcome the lack of one compliant offer by a single provider.

SLA negotiation with multiple Cloud providers by delegation to to a third party, represented by a broker in a market based context. To date, most research in service provision has concentrated on how to manage SLA compliance as well as tracking performance for planning purposes. There are many tools that help to measure, and track performance of service levels based on the actual service usage. The results obtained from such metrics are a necessary component in planning corrective actions. Automated contract creation enables service providers and their clients to make use of technology to create SLAs within pre-planned and pre-approved parameters [25].

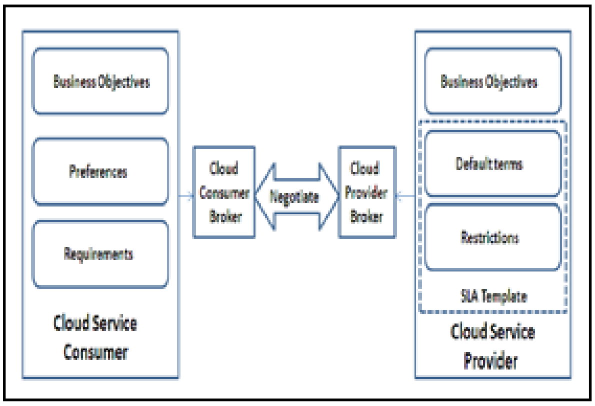


Fig. 5: SLA Negotiation Process

B. SLA Monitoring and measuring

1. Monitoring

of Cloud utilization can be delegated to a trusted third party 1) by providers who want maximize utilization of their resources in order to optimize profit, without violate the agreements; 2) by user, who has conflicting interests with providers

2. Monitoring

Providers monitor utilization of their resources for billing, to change bid prices in order to optimize profit, and to not exceed in resource allocation beyond the capability of respect the agreements. Users have conflicting interests with providers, needs to trust a third party that can be delegated to monitor the satisfaction of the agreed service levels [26].

Monitoring process should provide information about:

- Under-utilization of cloud resources, in order to negotiate cheaper agreements.
- Saturation of resources, to not let the users' applications work under the QoS level granted to users' clients.
- Unbalanced utilization of Cloud resources, in order to check the correctness of negotiated parameters, or to tune the execution of applications in the Cloud.
- Violation of SLA by providers.

Service level management, based on SLOs, is how performance information on the cloud is gathered and handled. This is how it is employed:

The cloud provider uses service level management to make decisions about its infrastructure; for example, if throughput isn't always meeting a customer's requirements, the provider can reallocate bandwidth or add more hardware. or decide to make one customer happy at the expense of another one. For providers, SLM is designed to help make the best decisions based on business objectives and technical realities [27].

The cloud consumer uses SLM to decide how he wants to use cloud services; like whether or not to add in more virtual machines and at what price point that option becomes too expensive to justify the return. For consumers, SLM helps them make decisions on the way they use the cloud [28].

C. SLA Mapper

The main task of the SLA Mapper is to create a link between the SLA elements and the knowledge of provider and customer at the end of each phase of the introduction process [29].

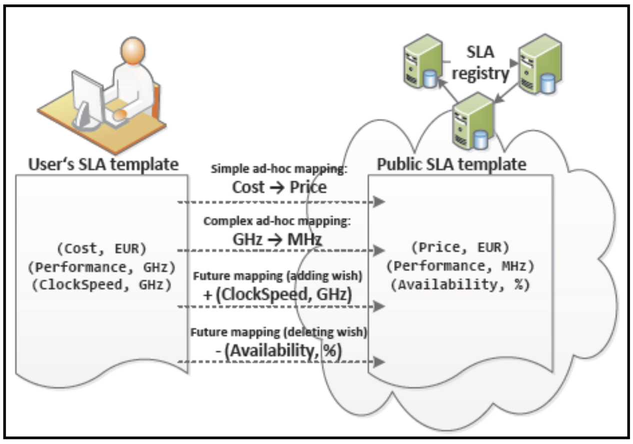


Fig. 6: Architecture of SLA Mapping Process

V. SLA Management

Service Level Agreements (SLA) are formal negotiated agreements that help to identify expectations, clarify responsibilities and facilitate communication between a service provider and its customer. We understand SLA Management as the management of service delivery systems in order to meet the QoS objectives (goals) specified in SLAs [30]. As SLAs are the core artifact for describing offered, requested or agreed service characteristics, the SLA lifecycle focuses on the steps of interaction between a service provider and a service customer. It includes the following stages:

1. SLA Template design ensures that offered QoS guarantees are realistic;
2. SLA negotiation ensures that agreed QoS guarantees are realizable;
3. SLA runtime ensures that QoS guarantees are satisfied;
4. SLA (Template) archiving ensures that previous experience is available to future cycles.
5. The management of SLAs happens in the context of the overall service lifecycle, which consists of the following stages:
6. Design and Development of artifacts needed for service implementation;
7. Service Offering prepares service artifacts for their instantiation and offering a service to customers;
8. Service Negotiation between customer and provider ideally results in an agreed SLA;
9. Service Provisioning creates an actual service instance which may include booking, deployment, and configuration activities;
10. Service Operations addresses an actual service instance that is up and running. It might be adjusted in order to enforce an SLA;
11. Service Decommissioning subsumes activities to stop a service instance so that it cannot be accessed by the service customer anymore [31].

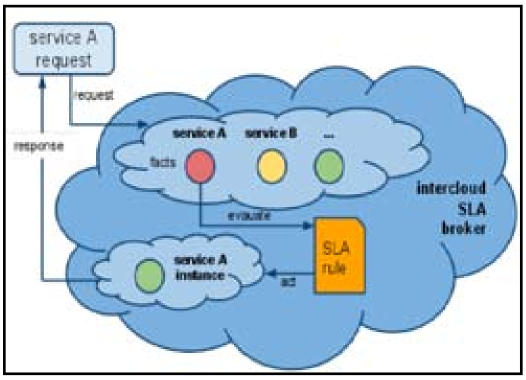


Fig. 7: Relation between Cloud Service and SLA

VI. Factors Needs to Consider Before Selecting a Cloud SLA

The following factors needs to consider when considering an SLA

A. Security

A consumer must understand his security requirements and what controls and federation patterns are necessary to meet those requirements. A provider must understand what they must deliver to the consumer to enable the appropriate controls and federation patterns.

B. Data Encryption

Data must be encrypted while it is in motion and while it is at rest. The details of the encryption algorithms and access control policies should be specified [32].

C. Privacy

Basic privacy concerns are addressed by requirements such as data encryption, retention, and deletion. An SLA should make it clear how the cloud provider isolates data and applications in a multi-tenant environment [33].

D. Data Retention, Deletion

How does your provider prove they comply with retention laws and deletion policies?

E. Hardware Erasure, Destruction

How does your provider prove they comply with retention laws and deletion policies [34].

F. Regulatory Compliance

If regulations must be enforced because of the type of data, the cloud provider must be able to prove his compliance.

G. Transparency

For critical data and applications, providers must be proactive in notifying consumers when the terms of the SLA are breached. This includes infrastructure issues like outages and performance problems, as well as security incidents [35].

H. Certification

The provider should be responsible for proving required certification and keeping it current.

I. Performance Definitions

What does uptime mean? All the servers on every continent are available? Or just one is available? It pays to define those definitions. (The authors of this paper suggest standardizing performance terminology to make it easier [36].

J. Monitoring

For issues of potential breaches, you might want to specify a neutral third-party organization to monitor the performance of the provider.

K. Auditability

Because the consumer is liable for any breaches that occur with loss of data or availability, it is vital that the consumer be able to audit the provider's systems and procedures. The SLA should make it clear how and when those audits take place. They can be disruptive and costly to the provider.

L. Metrics

These are the tangible something's that can be monitored as they happen and audited after the fact. The metrics of an SLA must be objectively and unambiguously defined. Following this list is a list of common metrics [37].

M. Providing a Machine-Readable SLA

This can allow for an automated, dynamic selection of a cloud broker. In other words, if your SLA requires that the broker use the cheapest possible provider for some tasks but the most secure provider for others, this type of automation makes it possible. (This type of service is not readily available yet, but is something to keep in mind when contributing to the cloud SLA standardization discussion [38].

N. Human Interaction

On-demand self-service is one of the basic characteristics of cloud computing, but your SLA should take into account that when you need a human being, one is made available to you.
Some of the common performance metrics
Throughput: System response speed.

O. Reliability

System availability.

P. Load Balancing

When elasticity kicks in.

Q. Durability

How likely to lose data.

R. Elasticity

How much a resource can grow.

S. Linearity

System performance as the load increases.

T. Agility

How quickly the provider responds to load changes.

U. Automation

Percent of requests handled without human interaction.
Customer service response times.

VII. Conclusion and Feature Work

In this paper, we introduced an approach for cost-efficient utilization of public cloud computing SLA templates in Cloud markets. we presented the Cloud SLA Life Cycle. In This paper we also given the Cloud Computing SLA negotiation process between service provided and consumer. In addition we well presented the SLA Architecture and SLA end to end implementation. We also described SLA mappings for market cloud players, allowing clients to utilize new public SLA templates. Furthermore, we investigated management of public SLA templates both by their structures and their SLO values.

In our future work, we will investigate the possibilities of reducing the cost of creating SLA mappings to the initial public SLA templates. We will do this by introducing a knowledge component managing the history of previous private SLA templates and SLA mappings, and apply machine learning methods to advise users of possible mappings to create. Furthermore, we will explore the methods for measuring market liquidity and adapt public SLA templates so as to increase the liquidity to its maximum point.

Reference

[1] Rabi Prasad Padhy, Manas Ranjan Patra, Suresh Chandra Satapathy,"X-as-a-Service:Cloud Computing with Google App Engine", Amazon Web Service, Microsoft Azure and Force.com. 2011, pp. 8-16.

[2] P. Kar¨anke, S. Kirn,"Service level agreements: An evaluation from a business application perspective, in eChallenges e-2007", IEEE-CS Press, 2007, pp. 104–111.

[3] L.Wang, J. Tao, M. Kunze, A. C. Castellanos, D. Kramer, W. Karl,"Scientific cloud computing: Early definition and experience", High Performance Computing and Communications, 10th IEEE International Conference on, pp. 825–830, 2008.

[4] Z. Wang, X. Zhu, S. Singhal,"Utilization and SLO-Based Control for Dynamic Sizing of Resource Partitions in DSOM", ser. Lecture Notes in Computer Science, J. Sch¨onw¨alder and J. Serrat, Eds., Vol. 3775. Springer, 2005, pp. 133–144.

[5] Z. Liu, M. S. Squillante, J. L.Wolf,"On maximizing service-level agreement profits", in EC '01: Proceedings of the 3rd ACM conference on Electronic Commerce, 2001, pp. 213–223.

[6] S. Shelford, M. M. Akbar, E. G. Manning, G. C. Shoja,"Distributed Optimal Admission Controllers for Service Level Agreements in Interconnected Networks", in Applied Informatics, M. H. Hamza, Ed. IASTED/ACTA Press, 2003, pp. 565–570.

[7] H. Demirkan, M. Goul, D. S. Soper,"Service Level Agreement Negotiation: A Theory-based Exploratory Study as a Starting Point for Identifying Negotiation Support System Requirements", In Proceedings of the 38th Hawaii International Conference on System Sciences, Big Island, HI, USA, January 2005.

[8] J.J.M. Trienekens, J.J. Bouman, M. van der Zwan., "Specification of Service Level Agreements: Problems, Principles and Practices", Software Quality Journal, 12(1), pp. 43–57, March 2004.

[9] Z. Shu, S. Meina,"An Architecture Design of Life Cycle based SLA Management", In Advanced Communication Technology (ICACT), 2010 The 12th International Conference on, Vol. 2, pp. 1351 –1355, Feb. 2010.

[10] I. Brandic, D. Music, S. Dustdar,"Service mediation and negotiation bootstrapping as first achievements towards self-adaptable grid and cloud services", in Grids meet Autonomic Computing Workshop 2009. In conjunction with the 6th International Conference on Autonomic Computing and Communications. Barcelona, Spain: ACM, 2009, pp. 1–8.

[11] M.B. Chhetri, J. Lin, S. Goh, J.Y. Zhang, R. Kowalczyk, J. Yan.,"A Coordinated Architecture for the Agent-based Service Level Agreement Negotiation of Web Service Composition", Proc. Australian Software Engineering Conference, 2006.

[12] E. Di Nitto, M. Di Penta, A. Gambi, G. Ripa, M. Villani. "Negotiation of Service Level Agreements An Architecture and a Search-Based Approach", Proc. ICSOC 2007, pp. 295-306.

[13] Keller, A., Ludwig, H.,"The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services", Journal of Network and Systems Management 11(1), pp. 57–81 2003.

[14] Sahai, A., Graupner, S., Machiraju, V., van Moorsel, A.,"Specifying and Monitoring Guarantees in Commercial Grids through SLA", In: 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2003), Tokyo, Japan, May 12-15, pp. 292–299, 2003.

[15] J. Spillner, A. Schill,"Dynamic SLA Template Adjustments based on Service Property Monitoring", Proc. of the 2009 IEEE Conference on Cloud Computing, 2009.

[16] A. Keller, H. Ludwig.,"Defining and Monitoring Service Level Agreements for Dynamic e-Business", In Proceedings of the 16th USENIX System Administration Conference, Philadelphia, PA, USA, November 2002.

[17] M. Risch, I. Brandic, J. Altmann,"Using SLA mapping to increase market liquidity in Service-Oriented Computing", ICSOC/ServiceWave 2009 Workshops, ser. Lecture Notes in Computer Science. Springer, 2010, Vol. 6275, pp. 238–247.

[18] M. Maurer, V. C. Emeakaroha, M. Risch, I. Brandic, and J. Altmann, "Cost and benefit of the SLA mapping approach for defining standardized goods in cloud computing markets", International Conference on Utility and Cloud Computing (UCC 2010) in conjunction with the International Conference on Advanced Computing (ICoAC 2010), 2010.

[19] Ching, A., Sacks, L., McKee, P.,"SLA Management and Resource Modelling for Grid Computing", In: London Communications Symposium (LCS 2003), London, UK, September 8-9, 2003.

[20] A. Korn, C. Peltz, M. Mowbray,"A Service Level Management Authority in the Cloud", Technical Report, 2009.

[21] S. Abdelwahed, S. Neema, J. P. Loyall, R. Shapiro,"A Hybrid Control Design for QoS Management in RTSS", IEEE Computer Society, 2003, pp. 366–370.

[22] N. Muller,"Managing Service Level Agreements", International Journal of Network Management, 9, pp. 155–166, 1999.

[23] J. Myerson,"Use SLAs in a Web Services Context", IBM developer Works, 2002-2005.

[24] Lodi, G., Panzieri, F., Rossi, D., Turrini, E.,"SLA-Driven Clustering of QoS-Aware Application Servers", IEEE Transactions on Software Engineering 33(3), pp. 186–197, 2007.

[25] F. Raimondi, J. Skene, W. Emmerich,"Efficient Online Monitoring of Web-Service SLAs", Proc. of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering, 2008.

[26] A. Keller, H. Ludwig,"The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services", Technical report, IBM Research RC22456, March 2003.

[27] R. Buyya, C.S. Yeo, S. Venugopal,"Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities", Proc. of the 2008 10th IEEE international Conference on High Performance Computing and Communications (HPCC.2008), IEEE Computer Society, 2008, pp. 5-13.

[28] Dobson, G., Sanchez-Macian, A.,"Towards Unified QoS/ SLA Ontologies", Services Computing Workshops, SCW '06, Sept. 2006.

[29] Y. Chen, S. Iyer, X. Liu, D. Milojicic, A. Sahai,"SLA Decomposition: Translating Service Level Objectives to System Level Thresholds", Int. Conf. on Autonomic Computing, 2008.

[30] D. Ardagna, M. Trubian, L. Zhang,"SLA Based Resource Allocation Policies in Autonomic Environments". J. Parallel Distrib. Comput., pp. 67:259–270, 2007.

[31] H. Li, G. Casale, T. Ellahi,"SLA-Driven Planning and Optimization of Enterprise Applications", In Proceedings of the first joint WOSP/SIPEW international conference on Performance engineering, WOSP/SIPEW '10, pp. 117–128, New York, NY, USA, 2010. ACM.

[32] SLA@SOI project: IST- 216556,"Empowering the Service Economy with SLA-aware Infrastructures", [Online] Available: http://www.sla-at-soi.eu/

[33] W. Theilmann, R. Yahyapour, J. Butler,"Multi-level SLA management for service-oriented infrastructures. Towards a Service-Based Internet", 2008, pp. 324–335

[34] M. Comuzzi, C. Kotsokalis, C. Rathfelder, W., U. Winkler, G. Zacco,"A Framework for Multi-level SLA Management", Proc.of 3rd Workshop on Non-Functional Properties and SLA Management in Service-Oriented Computing (NFPSLAM-SOC'09), November 23, Stockholm, Sweden.

[35] Menasce, D.A., Ruan, H., Gomaa, H.,"QoS management in service-oriented architectures", Performance Evaluation, 64 pp. 646–663, 2007.

[36] Bodenstaff, L., Wombacher, A., Reichert, M., Jaeger, M.C.,"Monitoring dependencies for slas: The mode 4 SLA approach", Proceedings of IEEE International Conference on Services Computing, 2008. SCC '08., pp. 8-11 July 2008, Honolulu, USA. (2008) pp. 21–29.

[37] Li, H., Theilmann, W., Happe, J., SLA translation in multi-layered service oriented architectures: Status and challenges", Technical Report 2009-8, Universit¨at Karlsruhe (TH), April 2009.

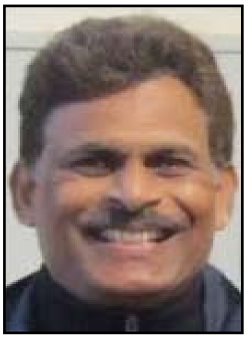
[38] Rabi Prasad Padhy, Manas Ranjan Patra, Suresh Chandra Satapathy,"Cloud Computing: Security Issues and Research Challenges", International Journal of Computer Science and Information Technology & Security, 2011, pp. 136-146.

[39] [Online] Available: http://www.rackspacecloud.com/legal/sla

[40] [Online] Available: http://aws.amazon.com/ec2-sla/

[41] [Online] Available: http://aws.amazon.com/s3-sla/

Rabi Prasad Padhy is currently working as a Senior Software Engineer - Oracle India Private Ltd. Bangalore. He has achieved his MCA degree from Berhampur University. He carries 8 years of extensive IT Experience with MNC's like EDS, Dell, IBM and Oracle. His area of interests include IT Infrastructure Optimization, Virtualization, Enterprise Grid Computing, Cloud Computing and Cloud databases. He has published several research papers in international journals. He is a certified professional for Oracle, Microsoft SQL Server database and also ITIL certified.



Dr. Manas Ranjan Patra holds a Ph.D. Degree in Computer Science from the Central University of Hyderabad, India. Currently he is an Associate Professor in the Post Graduate Department of Computer Science, Berhampur University, India. He has about 24 years of experience in teaching and research in different areas of Computer Science. He had visiting assignment to International Institute for Software Technology, Macao as a United Nations Fellow and for sometime worked as assistant professor in the Institute for Development and Research in Banking Technology, Hyderabad. He has about 90 publications to his credit. His research interests include Service Oriented Computing, Software Engineering, Applications of Data mining and E-Governance. He has presented papers, chaired technical sessions and served in the technical committees of many International conferences.

Dr. Suresh Chandra Satapathy, is a Phd in Computer science and Engg from JNTU Hyderabad. He has completed his masters from NIT, Rourkela in Computer science and Engg. His areas of interest are in Computational Intelligence, soft computing, machine learning, more specifically application of Swarm Intelligence techniques to data mining areas. He has about 60 papers in journals and international conferences like Springer, IEEE and Elsevier to his credit so far . He is the reviewer of reputed Elsevier journals and editorial board members of few international journals. He has edited two volumes of Springer proceedings from LNCS series and AISC series recently. He is a senior member of IEEE and member of CI society and SMC Society of IEEE and currently Secretary of IEEE CIS, Hyderabad section. Presently he is working as Professor and Head of Dept of CSE, Anil Neerukonad Institute of Technology and Sciences (ANITS), Vishakapatnam.