

# Service Level Agreement for implementing Ad Hoc Cloud Computing

Dr. Deepti Sharma  
 Professor, Department of IT  
 KCC Institute of Legal and Higher Education

**Abstract** - Cloud computing is a boon to the technological advancement which provides a flexibility and ensures a security to its users on some agreed upon services like Pay per use service, Software as a service, platform as a service and Infrastructure as a service. The need of service level agreement in cloud computing gives a framework for a defined set of condition from the both ends. This SLA gives assurance to its users. The service level agreement proposed in this paper provides freedom to employ the ad hoc cloud computing as it supports the security concern.

**Keywords:** Service Level Agreement, Pay per use, Service Provider, ad hoc, cloud computing

## I) INTRODUCTION

A Service level agreement basically provides a basis for ensuring the security to the organization. The concept of SLA marked its existence when service oriented industry came into picture. [1] SLAs are the legal documents and more accurately a signed document, which governs the delivery of services between the service provider and its clients, during the term of its relationships. An SLA provides the legal structure for relationship management between the service provider and client including contract monitoring, dispute resolution and remedies for breach penalties for sub-standard, late or non-delivery; contract change and termination. Service level agreements specify what is included and excluded in the services with the required terms and conditions to be delivered to the client. This is the most important aspect of service model which procures the consistency of relationship with the clients. It also helps the client to compare the service providers by using metrics and standards. SLA includes the additional cost for noncore services and remuneration for core services with the expenses incurred during the services. Any assets or resources provided to or transferred to the service provider must give in the SLA. In fact any ambiguity in the SLA design can lead to a big trouble in future if any one of the party is not true to its end. This can lead to defame the entire business image of client and provider if all the things are not well stated in the Service level agreement. For an effective SLA design, the levels of expected performance have to be realistically set: too low agreed service levels result in inefficient levels of support; too high agreed service levels result in contract frustration, and adversarial relationships. [5] An article on IBM website proposes the five levels in the service level agreement lifecycle that describes the states of the service level agreement life cycle, and, for each state, names the transition that moves a service level agreement forward to that state.

- Identification of SLA (Initial state) - This state is entered as soon as a consumer, represented by a capability version, requests a dependency on a service version or other capability version that offers the service level definition (SLD) that they require.
- Request for SLA - The agreed endpoints relationship target has been selected together with details of the required SLA properties and policies. The provider of the selected SLD must approve the request, reject it or ask for it to be revised.
- SLA request approval - The development team that want to consume the service can continue their development based on the consumption of this specific SLA, but they do not yet have authorization to access any endpoints.
- SLA activation - All the approved endpoints associated with the SLD, that are online, can be invoked using the terms of the SLA. There might be situations where the SLA is deactivated, in which case the SLA enters the SLA inactive state and any further interactions are blocked until it is reactivated.
- SLA termination - No interactions from this SLA are permitted.

## II) FACETS OF ADHOC CLOUD COMPUTING

Ad hoc cloud computing, allows cloud services to run on existing heterogeneous environment which is distributed over the network. This in turn improves organizations' resource utilization while offering some of the benefits of more conventional public and private clouds. This could yield significant cost savings. In the case of the various Cloud services (SaaS, PaaS, etc) the SLA will be different depending upon what you are paying for, e.g. Amazon Web Services guarantees a 99.9% uptime SLA or will compensate for time down. In particular, here the focus of interest is in increasing utilization of general-purpose computers in offices and laboratories. A proposed model here is given introduced by Graham Kirby, Alan Dearle, Angus McDonald and Alvero Fernandis in their research paper "An approach to cloud computing", [7] that describes the ad hoc cloud, in which infrastructure software is distributed over resources harvested from machines already in use. By ad hoc means that the set of machines comprising the cloud changes dynamically, as does the proportion of each machine's computational and storage

resources that can be harnessed at a given point in time. Thus, in contrast to the data center cloud model, resource provisioning levels are not established *a priori*, nor are resources committed exclusively to the cloud while in use. A participating machine is not dedicated to the cloud, but has some other primary purpose such as running interactive processes for a particular user, albeit often for a small proportion of the time. One of the most important research issues is how to reduce the impact of cloud operations on such processes to an acceptable level. The availability of ad hoc clouds could yield various benefits to individual enterprises. Firstly, it could reduce the numbers of machines that need to be purchased. Such costs are borne directly by enterprises employing private clouds, and indirectly by those using external cloud providers. The use of ad hoc clouds could also reduce the need for specialized infrastructure for resilience, such as redundant power and cooling systems, battery backup, etc. This represents 25% of data center costs [8]. Rather than ensuring resilience of a small number of physical buildings, the grain of resilience could be expanded by using more widely distributed machines and tolerating individual building failures. Following figure helps [12] to demonstrate the working of adhoc cloud that with the help of cloud local service provider distributes the service accordingly.

Ad hoc clouds could reduce overall power consumption. One factor is a reduction in the total number of machines required—significant since the energy cost of manufacture for a computer has been estimated as four times that used during its lifetime [9]. Another is that since machines comprising an ad hoc cloud infra-structure are situated in working spaces, the power consumed is partially offset (in temperate climates) by a reduction in the power required for

heating. Conversely, machines are housed at lower densities than in data centers, so less active cooling is required.

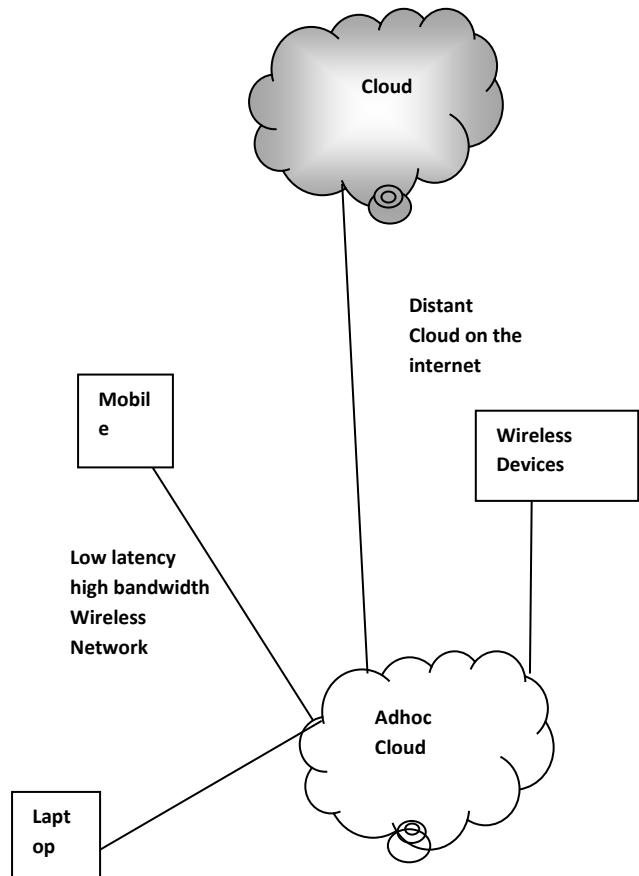


Fig. Functionality of adhoc cloud computing

### III) ROLE OF SLA FOR ADHOC CLOUD COMPUTING ARCHITECTURE

From late 2008 and early 2009, cloud comes so rapidly into pictures that service oriented industry look forward to it rapidly. In cloud computing a well defined SLA is must to include as an organization can not completely rely on service provider's data center for all its private data.[2]SLAs are particularly relevant to cloud

TABLE : State comparison between cloud and adhoc cloud [12]

Factor	Adhoc cloud	Cloud
State	Only Soft State	Hard/Soft state
Management	Self Management	Professional
Environment	Data Center in Box	Real Data Center
Ownership	Decentralized ownership by local business	Centralized
Network	LAN latency	Internet Latency/Bandwidth
Sharing	Few User	Huge Users

computing, an increasingly important and relevant deployment model for infrastructure, services, or platforms. Now for Adhoc cloud computing SLA plays even a more important role. Adhoc Cloud computing systems promise to offer subscription-oriented, enterprise-quality computing services to users worldwide wirelessly. With the increased

demand for delivering services to a large number of users, they need to offer differentiated services to users and meet their quality expectations. [4] Existing resource management systems in data centers are yet to support Service Level Agreement (SLA)-oriented resource allocation, and thus need to be enhanced to realize adhoc cloud computing. In addition, no work has been done to collectively incorporate customer-driven service management, computational risk management, and autonomic resource management into a market-based resource management system to target the rapidly changing enterprise requirements of adhoc Cloud. A book on service level agreement for cloud computing [2] describes its eight levels that includes 'Introduction to Service Level Agreements' that explores the role of SLA in Service oriented architecture, 'Foundations for Service Level Agreements', 'Scientific Innovations', 'Core Components of the Service Level Agreements Framework' have focus on service level agreement components, 'Management of the Business Layer', 'Management of the Software Layer', 'Management of the Infrastructure Layer', and 'Selected Business Use Cases' gives the actual implementation of SLA for cloud. But as the need of hour realizes the actualization of adhoc cloud, the stress should be given on SLA for Adhoc cloud.

#### IV) CHALLENGES AND REQUIREMENTS FOR SLA IN ADHOC CLOUD

There are certain issues concerning the requirement for service level agreement in adhoc cloud given hereunder.

Table 2: Requirement of SLA in adhoc cloud

Issues	Challenges	Requirement
Managing Customer end	[6] Customer satisfaction	<ul style="list-style-type: none"> <li>-&gt;Continues Communication to access the need of customer</li> <li>-&gt;Security measures undertaken against risks and doubts</li> </ul>
Managing Computational Risk	Risk Management	<ul style="list-style-type: none"> <li>-&gt;Establish the context</li> <li>-&gt;Identify the risks involved</li> <li>-&gt;Assess each of the identified</li> <li>-&gt;Risks, Identify techniques to manage each risk</li> <li>-&gt;Create, implement, and review the risk management plan.</li> </ul>
Managing resource allocation	Resource Management	<ul style="list-style-type: none"> <li>-&gt;Providers require autonomic resource management to selectively choose the appropriate requests to accept and execute depending on a number of operating factors</li> </ul>
SLA-oriented Resource Allocation Through Virtualization	Virtualization	<ul style="list-style-type: none"> <li>-&gt;VMs may be assigned various resource management policies</li> <li>-&gt;Better support the implementation of SLA-oriented resource allocation.</li> </ul>

#### V) FACTORS NEED TO BE CONSIDERED WHILE DESIGNING SLA FOR ADHOC CLOUD COMPUTING

As far as the Adhoc cloud computing is concerned whenever an organization leads to deal with its client, it must prepare a well defined SLA so that the client fully rely on the service provided by the provider. There are some important factors needs to be added to the SLA of adhoc cloud so that the mitigation

A) *Protocol Engine*: I have made a design choice to separate between the protocol (i.e. message exchange sequencing and control), and the negotiation strategies which concern decision-making as regards what constitutes a "good" SLA.

B) *Template Registry*: For instance, a template would include information to indicate that, it is possible to negotiate property Completion Time for a particular operation of a service. The domain-specific decision making algorithms know by design how to use this property, as long as the service provider offers it for negotiation. The template registry is a query-able data store, where I implement complex reasoning for matching queries (also implemented as SLA templates) to stored information.

C) *SLA Registry*: SLAs are stored in this registry after they are established. Queries return not only SLA information, but also the dependencies of one SLA onto others. This way, it is possible to find which SLAs would be affected if one fails.

D) *Service Manager Registry*: This data store is used by SLA Managers to manage the known Service Managers. It contains only static information about Service Managers; dynamic information (e.g. resource availability) is retrieved directly from the Service Manager using respective query methods.

E) *Monitoring Manager*: The Monitoring Manager (MM) determines the monitoring infrastructure that must be deployed in order to monitor a service, for which I'll establish one or more SLAs. In addition, the MM can perform monitorability checks, based on which it is possible to know at negotiation time, whether a SLA can be monitored at all. The rational is that a SLAM should never establish an agreement that it cannot monitor.

F) *Service Advertisements*: a Publish/Subscribe system is used to advertise new templates as they become available. Prospective customers can receive the advertisements, filter them, and cache them as initial indications of candidate providers for the services of interest. Follow-up queries to the respective providers' registries retrieve additional information about particular offers. I have not tied the system by design to a specific broadcasting technology; this is left open to the implementation.

Now I here focused on the above issues and realized certain additive parameters should be included as an essential feature of adhoc cloud SLA.

#### VI) CONCLUSION

This paper presents the Service Level Agreement as the most important doc to be signed by both the ends to concur the healthy professional and business relations. Further as compared to the SLA for cloud computing there must be

some additions done in the agreement. I have done an analysis on SLA about adhoc cloud computing where the core data and information of a client is at the data centre and that also passing through the wireless mode of communication. As while working in the wireless medium, security is the major concern and same is applicable to the cloud itself. By combining both one can easily understand the issue of security. That is why it is almost indispensable to discuss the SLA for cloud.

## VII) REFERENCES

- [1] An article on Hariott Watt University website
- [2] A book “Service Level Agreements for Cloud Computing” by Jessica McCarthy for Springer © Springer Science+Business Media, LLC 2011
- [3] Sample Service Level Agreement for Supply of ICT Product or Support Services between Agency Name And Company Name
- [4] *SLA-Oriented Resource Provisioning for Cloud Computing: Challenges, Architecture, and Solutions* by Rajkumar Buyya<sup>1,2</sup>, Saurabh Kumar Garg, and Rodrigo N. Calheiros
- [5] Article “ Service level agreement life cycle” on [http://publib.boulder.ibm.com/infocenter/st/v6r3/topic/com.ibm.sr.doc/rwsr\\_gep\\_sla\\_life\\_cycle.html](http://publib.boulder.ibm.com/infocenter/st/v6r3/topic/com.ibm.sr.doc/rwsr_gep_sla_life_cycle.html)
- [6] C. S. Yeo and R. Buyya. Integrated Risk Analysis for a Commercial Computing Service. *Proceedings of the 21st IEEE International Parallel and Distributed Processing Symposium (IPDPS 2007)*, Long Beach, CA, USA, March 2007
- [7] S. Garfinkel. An evaluation for Amazon’s Grid Computing Services: EC2, S3, SQS. Technical Report TR-08-07, School for Engineering and Applied Sciences, Harvard University, Cambridge, MA, July 2007.
- [8] Amazon’s Cloud Storage Hiccups. <http://www.wjla.com/news/stories/0208/496511.html>.
- [9] Foster, I. and Kesselman, C. The Globus Project: a Status Report. *Future Generation Computer Systems*, 15, 5-6 (1999), 607-621.
- [10] Service Provider Strategies for Building Trust by Steve Lesem
- [11] M. Satyanarayanan, P. Bahl, R. Cáceres, and N. Davies, “The case for VM-based cloudlets in mobile computing,” *IEEE Pervasive Computing*, vol. 8, no. 4, pp.14-23, Oct. 2009.
- [12] “A Reference Architecture for Multi-Level SLA Management” W. Theilmann, J. Happe, C. Kotsokalis, Edmonds, K. Kearney, J. Lambea