

# SLA Automated Negotiation Manager for Computing Services

Halina Kaminski, Mark Perry

Computer Science Department, University of Western Ontario

{hkaminsk | markp@csd.uwo.ca}

## Abstract

*Success in today's marketing arena can often depend on companies embracing effective new technologies and integrating them into their business model. In the computing service supply industry, Service Level Agreements (SLAs) are commonly prepared and signed agreements between the service provider and its customers. SLAs should match business needs of both sides of the agreement as closely as possible. This paper focuses on at the steps and activities that the service provider can take to facilitate agreement. It proposes an automated way for creating SLA's from a set of Service Level Objectives (SLOs). The SLA should achieve business goals, including the maximization of customer satisfaction. To automate the preparation of effective SLAs each company should set SLOs that support business needs.*

## 1. Introduction

The rapid improvement of web based technologies has expanded the potential for networked computing services. Both challenges and opportunities presented by information systems require highly competitive businesses that maximize their existing resources whilst simultaneously investigating new and better ways to move products and services to the forefront of the web-services marketplace. A key component for success in today's marketing arena is for companies to broaden their vision and embrace effective new technologies into their business model. Earlier work in SLA management has focused on a bottom up approach, looking to capture manage SLA data.[1] However, our work concentrates on automatic SLA creation that integrates an effective negotiation process. Currently most such negotiations are done offline and in a presence of executive personnel. One area in which companies are seeing increased cost is

that of personnel supporting their system offerings. Through the use of automation, such cost loads can be reduced.[2] The development of an SLA real-time negotiation system will provide a great asset to service provision enterprises.

In most information technology service offerings, the critical components and the level of delivery must be identified and agreed upon. Such steps must include development of SLAs between a service provider and its customers. SLAs can also be seen as Electronic Contracts for computing services when they are formed online. Every SLA is prepared and 'signed' by all parties involved in an agreement: this ensures that customers will get the service that they pay for, and defines the limits of obligations placed on the providers.

## 2. Service Level Agreements

As we are concerned with the generation of the SLA, here we look at the steps and activities of the service provider, as this is where the SLAs usually originate. From the provider's point of view the SLA should achieve its business goals; at the same time it should maximize the customer's satisfaction and minimizing liability for non-performance. SLA development should be considered as a vital step in the business process. The most recommended strategy to prepare good SLAs is to set level objectives that support business needs.[1] SLAs are more complex when the provider offers multiple services such as networking, online databases or end user direct support. [3] Usually, the services provided by an IT company vary both in diversity and complexity. To increase performance, forward-thinking organizations have implemented service level objectives (SLOs) to track the effectiveness of service infrastructure.

## 3. Service Level Objectives

Defining service levels for the strategic business services has been strongly recommended by marketing

experts. Every single resource that is offered to the customer should have an indication what its business levels are and what performance is acceptable to the end-user. The following are some guidelines to be considered when setting service levels. It is very important for SLOs to be realistic. The target goals have to reflect reality and should be attainable. They also should include the metric definition which contain how the values are measured and reported to the managing authority. Each SLO has to have a meaningful description of the service level such that it can be easily understood by a customer. For example, expressing the service performance in pockets dropped or server congestion may not mean anything to the end-user. Most importantly SLOs have to be cost effective. There is a belief, that the best SLOs are impractical, because they are very expensive to be measured. [4] To summarize, a good SLO should be realistic, should have quantifiable availability and performance that reflect business goals, and have defined good, yet not too complicated, means of measurement. Nevertheless, having objectives alone is not enough to provide a high quality services.

It is our goal to be able to set service levels for the resource (service) in such a way that they are not custom made but predefined and reusable. Ideally there should be many levels for the same resource and the levels would differ in QoS and the cost. Levels of service can be pre-defined for the resources of the same type and the same level of service can be used by many customers. Service level objective also expresses a commitment to maintain a particular state of the service in a predefined period of time.

The existence of a number of service levels and performance metrics for each resource results in multiple SLOs for every service.

## 4. Negotiations

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. Our research is aimed at the automation of SLA development and creation, which is synonymous with electronic contracts for computing services, from SLOs. In addition to giving flexibility to the contracting system will optimize the provider's profits at the same time as maximizing the customer's

satisfaction. We are developing a negotiating tool (SLA Negotiation Manager) described hereafter along with the process of negotiation and creation of an SLA from existing business objectives. We would like to provide an interactive negotiation system that would help a service provider to formulate and evaluate an offer, and then send that offer to the client.

Our system will be built in such a way that it will negotiate on behalf of service provider. The overall negotiation process will be modeled as exchanging proposals and counter-proposals between the provider and the customer. At the beginning provider needs to come up with an offer pack which is based on service, price, delivery, quality etc. For that a set of SLOs can be used. The provider takes all factors into account and calculates the expected pay-off value function associated with possible offers, and selects the offer that maximizes his payoff. If the customer is satisfied with an offer, he just sends an acceptance message to the provider and an SLA is finalized. If the client does not accept the first offer, then he can either abort the negotiations or he can send a counter - proposal. At this point the provider evaluates an offer and updates its knowledge about the customer.

If the offer is acceptable the Negotiation Manager creates an SLA, otherwise provider sends counter-proposal. Exchange of counter-proposals goes until the time allocated for the negotiation expires, or one of the parties decides to accept an offer or quit.

## 5. Implementation

An automated SLA creation system gives the opportunity to eliminate many of the inefficiencies caused by lack of resource specific knowledge. By using templates and SLO libraries SLA Negotiation Manager will ease the contract creation. We have adopted here the widely approved contract language Web Service Level Agreement (WSLA). The system will provide a user friendly interface for the client to see and choose requested services. It is anticipated that the contract creation time will be reduced significantly as a result of the usage of templates and pre-approved clauses. By using the system the service provider will ensure consistency and compliance with company's standards. Storing all SLAs in a single repository will provide an additional benefit to the service planning and management tools that will have to search for a contract in only one place. In the SLA creation process, a client is presented with the services that are offered by the provider. Based on the customer's choice the Negotiation Manager aggregates and combines them into various SLA parameters, chooses service levels (SLO) for every SLA parameter. Because every SLA

defines the agreed level of performance for a particular service it has to be checked for the resource availability (this process is also known as SLA compliance monitoring). The SLA Negotiation Manager embodies the business knowledge, goals, and policies of the party it belongs to. Such knowledge enables the system to choose and combine the set of SLOs that should be specified in the SLA in order to ensure compliance with the business goals.

In [5] it is shown that there are five main components of an enterprise Contract Lifecycle Management strategy:

- automated contract creation,
- secure contract negotiation,
- electronic contract repository,
- automatic upload of relevant contract data to back-end systems,
- generation of proactive management reports and alerts to encourage compliance to committed contract terms and conditions.

The SLA Negotiation Manager will fulfill first four out of these five directives. The system will automate contract creation through a secure negotiation with the customer, then newly created SLA will be stored in a central repository and the back-end system logs will be updated for the usage of resources that are specified in the contract. As for the last component, we leave the generation of relevant reports to the service management tools.

As an illustration of one type of situations, hereafter is a typical scenario:

A customer finds a service description and relative URL in the business directory (e.g. UDDI). Then he connects to the company that offers the service. Upon such connection an SLA Negotiation Manager is started. The customer wants to subscribe to a particular service (for example: disk space to store some back-up files). The customer knows that he needs to rent two Terabytes of space and is willing to pay \$100/month for it. The SLA Negotiation Manager by examining existing SLOs and existing SLAs checks if such service is available (checking of the existing SLAs is done in order to avoid over-commitment). If the provider's company can provide a service required then an SLA is created accordingly and presented to the customer for an acceptance. Upon customer's acceptance the SLA is stored into the repository and the service is made available to the client. *(It is anticipated that at this point an SLO defining a service of renting 2 Terabytes of memory would have to be removed from a pool of SLOs to avoid over-commitment.)* If the initial resource examination returns a negative response saying that the company can not commit to such service, then the SLA Negotiation Manager would come up with the next

best offer. Such decision making might be based on asking customer how much money he is willing to spend or how much memory he must absolutely have and based on that and on knowledge of the system the Negotiation Manager can propose a number of options to choose from. One option might be an offer of 1.5 Terabytes of storage for \$80.00/month and another offer could be 2 Terabytes of storage on two separate disks for a total of 120.00/month. Ideally the customer chooses one of the offers and an SLA is created. If the customer does not agree to the proposed service then negotiation continues.

### Negotiation Manager Model

An Automated Negotiation Manager model is a 7-tuple:  $\{R, K, Z, P, Q, F, M\}$  where:

**R** is a set of participants,

**K** is a set of all possible agreements (SLAs), **Z** is a set of business rules,

**P** is a set of all SLOs,

**Q** is a set of all negotiation sequences ,

**F** is a utility function,

**M** is a set of all possible offers.

**R** is a set of participants. This set contains all parties that can be involved in the contract. The customer, service provider and all supporting parties belong to this set. At least two elements of this set (service provider and customer) must participate in any SLA negotiation process  $q_n \in Q$ .

**K** is a set of all possible agreements (SLAs). Every existing SLA agreement that is stored in a data base belongs to the set K. It also contains all the possible agreements that can be created as a result of any successful negotiation process.

**Z** is a set of business rules (also called business knowledge). A business rule that a service can not cost less than \$0.07 per transaction might be an example of  $z_i \in Z$ . Set Z represents corporate preferences and aligns business strategies of a service provider.

**P** is a set of all SLOs. Every SLA contains at least one SLO for the agreed service.

**Q** is a set of all sequences  $s$ , such that every  $s = q_1, q_2, q_3 \dots q_n$  where  $q_i$  is an action (an offer, a counteroffer, accept or decline). Each  $s$  illustrates a negotiation process and every successful negotiation is a finite sequence  $s$ . Here, by successful negotiation we mean any negotiation process that resulted in either accept or decline. Sequence  $s$  can also serve as a history log when stored in a repository. The past negotiation procedure can be recreated from such sequence.

**F** is a utility function. This function is customized according to the negotiating party needs and business preferences. For example it might be widely known that the customer offers 10% less for the service than

he is really willing to pay. Function  $f$  might be used to calculate next offer:  $f = \text{current offer} * 10\%$ .

$M$  is a set of all possible offers. Every permutation of elements of  $P$  belongs to  $M$ . In addition  $M$  contains any combination of an offer that has been modified according to one or more business rules from set  $Z$ .

There have been many mathematical models developed for negotiations, typically on direct e-commerce negotiations, and often employing game theory algorithms. [6,7] Although these are not directly applicable to the SLA environment where there are a great deal more factors to consider above the product and price, they are useful for further development of the negotiation system.

In order for the automated SLA Negotiation Manager to work, the SLOs have to be translated into the machine readable format. There are several such specifications resulting from ongoing research at the large software companies such as HP, Sun Microsystems and IBM. [8,9] For our model we have chosen WSLA expressions to express the SLOs. WSLA language was developed in 2003 by the researches from IBM, and the description for the specification parameters has been published in [4,10]. For the purpose of our research we assume that SLOs are developed by the Business/Marketing department and are already defined in WSLA.

There are many different types of SLAs, ranging from the very basic to precise, focused SLAs that vary from customer to customer in the same enterprise.[3] Each customer, no matter how large or how small of an enterprise, has to define thresholds for acceptable service performance for the price he is willing to pay. On the other hand, the service provider has to decide whether the SLA's requirements are acceptable and if the client-specified service performance needs are not too restrictive. In our automated model for SLA creation, the system will provide the compliance monitoring according to the customers choices.

In our model, every customer no matter how small or how large of an enterprise will be able to take advantage of an automatic SLA creation through our SLA Negotiation Manager. The resulting SLA will be based on the SLOs of the business, and created according to WSLA specifications, which in turn will make them readable for other system utilities such as performance manager or service level manager.

## 6. Conclusion

In practice constructing an SLA requires planning and care. While the process can vary among companies, it is often a politically oriented topic: SLAs are known to be used to find blame instead of being a driving force towards a positive change. This is mostly

because the development process for SLAs has failed to keep up with the rapid changes in technology.

The SLA Negotiation Manager will provide an automated way to create and document SLAs that will increase service provider's profits, maximize customer satisfaction, and it will open up the way to more flexible service provision.

## References:

- [1] Bucu M.JU., Chang R.N., Luan L.Z., Ward C., Wolf J.L., Yu P.S. "Utility computing SLA management based upon business objectives" IBM Systems Journal Vol 43 No.1 2004 p.159.
- [2] Suh, Bob. "Avoiding an Austerity Trap" Outlook Journal, February 2004 Retrieved from: [http://www.accenture.com/Global/Research\\_and\\_Insights/By\\_Subject/High\\_Performance\\_Business/AvoidingtheAusterity\\_Trap.htm](http://www.accenture.com/Global/Research_and_Insights/By_Subject/High_Performance_Business/AvoidingtheAusterity_Trap.htm) on Dec 12, 2005
- [3] Leopoldi, R. "IT Services Management, A Description of Service Level Agreements", White Paper, RL Consulting, 2002 Retrieved from: <http://www.itsm.info/SLA%20description.pdf> on June 22, 2005
- [4] Sturm, Richard. "Service Level Objectives", Network Word Fusion, 2002 Enterprise Management Associates, Inc. Retrieved from: <http://www.slm.info.com/articles/slobjectives.htm> on Dec 12, 2005
- [5] Weintraub Allan, "Contract Management – A Strategic Asset" CRMToday website , Retrieved from: <http://www.crm2day.com/highlights/EEpIVVVFlpFCMLrUcN.php> On June 22, 2005
- [6] Zeng, D., and Sycara, K. "Bayesian Learning in Negotiation" Working Notes of the AAAI 1996 Stanford Spring Symposium Series on Adaptation, Co-evolution and Learning in Multiagent Systems
- [7] Oprea M., "An Adaptive Negotiation Model for Agent-Based Electronic Commerce", Studies in Informatics and Control, Vol.11, No 3, September 2002
- [8] Dan, A., Ludwig, H., Pacifici, G., "Web Service Differentiation With Service Level Agreements", White Paper, IBM Corporation, March 2003, Retrieved from: <http://www-106.ibm.com/developerworks/library/ws-slafram/> on Feb 02, 2005
- [9] Sun Microsystems, "Using the Sun ONE Application Server 7 to Enable Collaborative B2B Transactions" Informit Network Website, Retrieved from: <http://www.informit.com/articles/article.asp?p=100664&seqNum=2&rl=1> on Feb 22, 2005
- [10] Keller, A, Ludwig, H., "The WSLA Framework: Specifying and monitoring Service Level Agreements for Web Services", Journal of Network and Systems Management, Special Issue on E-Business management, Volume 11, Number 1, Plenum Publishing Corporation, March 2003.