Verity: A QoS Metric for Selecting Web Services and Providers

Sravanthi Kalepu, Shonali Krishnaswamy, Seng Wai Loke,
School of Computer Science and Software Engineering, Monash University, Australia
sravanthihere@hotmail.com, shonali.krishnaswamy@csse.monash.edu.au,
swloke@csse.monash.edu.au

Abstract

With the proliferation of web services, quality of service serves as a benchmark to differentiate the services and their providers. As of today, a wide spectrum of attributes have been identified to account for the quality of a service like availability, reliability, servability, performance, reputation and so on. Reputation has been measured as an average user rating and we argue that the user perception alone is not sufficient to indicate the reputation. It is necessary to measure how trustworthy the provider has been in complying with the agreed levels in the SLA. To quantify the consistency in compliance levels, we introduce a new QoS attribute termed verity and propose an architecture to quantify it. We argue that verity should be taken into account for a quality driven selection and composition of web services. Reputation, when expressed as a vector of user rating, compliance and verity is a more intuitive indicator of the provider's trustworthiness.

1. Introduction

Web services are evolving as an innovative mechanism for rendering services to arbitrary devices over the WWW [20, 10]. They are interoperable across platforms and neutral to languages, which makes them suitable for access from heterogeneous environments. As a consequence of the rapid growth of web service applications and the abundance of service providers, the consumer is faced with the inevitability of selecting the "right" service provider. In such a scenario the quality of service (QoS) serves as a benchmark to differentiate service providers. QoS has typically been associated at a network/server level rather than the user level [5]. As the Internet expands the diversity of its applications these issues become increasingly critical and it is now necessary to relate the objective system quality to the users' subjective perception of quality [5].

Placed in the context of web services, quality of service comprises of techniques that aim to bring a balance between the needs of the service requestor and those of the service provider while being constrained by the limited network and server resources [4]. QoS has been the distinguishing aspect in the works addressing quality aware UDDIs [1], distributed query processing [2], real time quality measurements [9], composition [8] and selection [22] of web services.

The interface definition language (WSDL) in the web services architecture describes only the syntactic properties of the web service and does not specify QoS parameters. QoS enabled web services are therefore typically associated with a service level agreement (SLA) which is used to guarantee quantifiable performance. It is a formal binding defining the relationship between the service provider and the customer often involving a third party to support the enactment of the contract [6]. It consists of descriptions of the parties involved, the service definitions, quality of service parameters and their service levels and possible penalties in the face of failure to comply with the agreed levels [7]. SLAs in web services have been developed to facilitate the complex relationships among providers and guarantee application performance [6].

Several quality attributes like *response time, availability*, *performance* and *throughput* are regarded the criteria for quality driven selection of web services [8, 2, 1] and typically are included in the SLA as part of the service definition. Strongly amiss among them is an attribute that measures how often the service provider has accurately delivered the agreed levels of these attributes. There is no mechanism to quantify how trustworthy the service provider has been in previous service deliveries. The degree of trustworthiness has been loosely defined as the reputation of the provider in [8]. The authors measure reputation to be the average ranking given by end users. This represents only the users' perspective of the provider and does not capture the degree of variance in the service providers' compliance levels. There will be occasions when the agreed service levels are adhered to and others when penalties are incurred. There is a need to aggregate these positive and negative levels to indicate the net compliance value. The compliance profile of the service provider will be used in selecting the “right” provider and there is a need to differentiate the local and global profiles of the provider. The end user is entitled to a personalized regard to the providers' services, which is not necessarily in concurrence with the world view. In the event of the global profile being unavailable, the end user should still be able to employ the local profile to aid in the selection process. The end user should therefore be given the flexibility to exercise personal regard and override the global prospect as the needs may be.

Traditional QoS metrics like response time and delay no longer suffice to fully describe the quality of service as perceived by users [5]. In the light of these issues, this paper introduces a new attribute *verity*, to be an indicator of the providers' truthfulness. Verity of a web service provider is measured by external components and is defined as the ability to maintain the lowest difference between the projected and achieved levels of service metrics. *Reputation* is defined as a vector of verity, ranking or end user perception and compliance levels. This enables *reputation* to be a more indicative measure of the quality of service. Quality driven web service selection should therefore include verity along with the more widely used attributes like *response time, availability, performance* and *cost* to deliver high-end services to the end users.

The remaining of this paper is organized as follows. Section 2 identifies the various QoS metrics from contemporary QoS research in web services and discusses related work in reputation mechanisms. Section 3 introduces the attribute *verity* and its relation to the *reputation* of the service and service provider. Section 4 explores the architecture proposed to integrate *verity* with the QoS stream. Section 5 formalises the *verity* calculations. Section 6 summarises and concludes the paper.

2. Related Work

The significance of QoS in the performance of the Web is evident from on-going research. Efforts have been concentrated in identifying the QoS attributes [1,2,4,8], improving the quality of service through replication of services [2], load distribution [13] and server redirection [24], measuring the QoS attributes like *response time* [4], *availability* [23] and improving the response time of web pages [15]. The work in [9] quantifies various attributes of QoS along with presenting an adaptive control mechanism for guaranteeing the service levels. QoS is extended with the concepts of quality of experience (QoE) and quality of business (QoB) in [14] which are claimed to be increasingly important in quantifying the user experience and business return. Quality driven selection and composition of multiple web services to constitute a single task is the subject of the work in [8] and a multi-attribute utility theory and linear programming-based methodology have been developed to address the problem. These works fortify the argument for quality of web services and indicate the areas to be further explored.

2.1 QoS Attributes

A wide spectrum of metrics which attribute to quality of service has been put forth by the research community with often varying interpretations. Presented here is a list of these metrics with multiple definitions where applicable.

Availability: Availability is the quality aspect of whether the Web service is present or ready for immediate use [4] represented as the percentage of uptime of a service in an observation period[9,2,1,8] and related to its reliability[1]. For frequently accessed services a small value of the observation period gives a more accurate approximation for its availability [8].
Reliability: It is the probability that a request is correctly responded within a maximum expected time frame or simply the success rate of finish [8,9]. It refers to the assured and ordered delivery for messages being sent and received by service requestors and service providers [4].
Price: The monetary value of the service as set by the service provider [8].
Throughput: Throughput is the number of web service requests served in a given period of time [1]. The response time of a system increases as the throughput increases and an important policy decision is to make a compromise between maximizing the throughput and minimising the response time [2]. QoS measures can include the maximum throughput or a function that describes how throughput varies with load intensity [10].
Response time: The amount of time between sending the request and receiving a response [2] or the guaranteed average time required to complete a service request [1,9]. Also referred to as execution duration, it is computed using the processing time and the transmission time [8]. It is sometimes referred to as latency [4].
Latency: Time taken between the service request arriving and the request being serviced. The throughput of a system is related to its latency [1].
Performance: It is measured in terms of throughput and response time. The service is considered to perform well when the throughput is high and has a low response time [4]. It is the speed in completing a service request and is

indicated by response time, latency and throughput [1].

Security: The ability to ensure non-repudiation of messages, confidentiality and authentication of concerned parties [4]. It is also indicated by the traceability, accountability, ability to encrypt data [1] and resilience to denial-of-service attacks [10].

Accessibility: The quality aspect of a web service that represents the degree to which it is capable of serving a request. It is the chance of a successful service instantiation at a point in time [4].

Regulatory: The quality aspect that shows the conformance of the web service with the law, rules, compliance with standards and the established service level agreement [4].

Robustness/Flexibility: The ability of a service to withstand invalid, incomplete or conflicting inputs and produce the correct result [1].

Accuracy: It is the rate of errors produced by the service [1].

Servability: The percentage of time the server can accept a service request and is lowered when the system's response time increases and the server does not accept any requests [2]. This attribute is closely related to accessibility in [4].

Integrity: The ability of the web service to maintain the correctness of the interaction which is ensured by the proper execution of a web service transaction [4]. This idea is also expressed in [1] as transaction support related attribute of QoS.

Reputation: It is a measure of the trustworthiness of a service and depends on the end user's experiences of using it. It is the average ranking given to the service by the end users [8]. Reputation is further explored in the following sections of this paper.

2.2 Reputation in Web Services

Reputation is indicative of the confidence and trust placed in a system or entity's ability to deliver desired results. It also represents the user perception and is conceived as a multidimensional attribute since the same system can be reputed differently in different domains [11]. A reputation system is intended to aggregate and disseminate feedback about participants' past behaviour [12]. These systems encourage trustworthiness and help people choose the right system for service request.

Reputation has been an area of intense activity in the e-market place and web pages realm. A manual reputation gathering system is used in the news website Slashdot to rank articles thereby screening out articles judged to be of lower quality [16]. Weaving a Web of Trust creates a personal recommendation system for web pages where the ratings given by users to themselves are verified by a central agency or other trusted users [17]. Collaborative filtering is a popular technique used to build reputation systems and is implemented in Firefly [18]. The technique works by detecting meaningful patterns from data pools collected from different users and makes recommendations to people based on others who showed a similar taste. E-commerce web sites like Amazon, e-Bay [19] and OnSale Exchange [20] also use reputation mechanisms which are based on aggregating the numeric rating given to the concerned parties and optionally including a textual note about the performance.

In the context of web services too, reputation is defined as the average of rankings given to a service by all users who invoked and rated the service [8]. The authors in [11] have developed an improvised reputation system where reputation is built up over its historical values, does not exceed a maximum value and is a weighted average of non-negative ratings. More weighting is given to the rating given by users who themselves have a good rating. The same work also describes a more personalised reputation system that uses directed graphs with the edges representing the most recent reputation rating given by one user to another. Whenever a directed path exists between two users, a more personalised reputation value can be computed for them [11].

An agent based architecture for reputation and endorsement of web services has been developed by the authors in [3]. In this architecture, a Web Service Agent Proxy (WASP) conveys reputation data that is delivered by human intervention or provided by agents, to a Reputation and Endorsement Agency (RES) where it can be aggregated. The rating of a service is a vector of attribute values and the reputation is modelled as a vector of aggregate ratings of the attributes from different attribute groups.

This paper takes an extended approach to reputation. Although we concur with the above works that reputation is chiefly influenced by the user perception, it should also be an indicator of the trustworthiness and consistent compliance levels. To be an indicator of trustworthiness, we introduce a new attribute of quality termed *verity* and propose a formal method to quantify it. It is suggested that *verity* should be an important contributor to the quality driven selection and composition of web services and deserves focused attention.

3. Verity and its impact on Reputation

The current trend in identifying a good quality service provider lies solely in assessing the quality parameters supplied by the providers themselves. There is no provision for a quality attribute, which keeps track of the provider's previous performance and assesses the compliance with the stated levels of quality attributes in the SLA. The quality attribute *verity* proposed in this paper refers to the degree of variance in the compliance levels of a service provider and delivers a mechanism for assessing the reputation of the provider.

The concept of verity stems from the fact that there has been no practice of recording the achieved service levels once a transaction has been completed. Doing so gives an insight in to the providers past performance by providing necessary data to progressively assess the compliance levels over a range of past transactions. Compliance refers to the service provider's ability to meet the service level of each QoS parameter laid out in the SLA without incurring penalties. The difference in the actual and projected values is ideally zero and is best kept to a minimum to achieve high compliance levels. Compliance is calculated as the mean of the normalized difference in the projected values of a QoS parameter and the actual values.

The variance in the compliance levels is a direct indicator of the service provider's ability to consistently deliver guaranteed levels of service and is therefore a measure of verity. Lower the variance, the more successful is the provider in delivering consistent service levels. Verity is measured over a range of compliance values obtained from past invocations of the service. The concerned calculations of verity are formalised later in the paper.

The closest contemporary research has come to assessing the trustworthiness of a service provider is to measure the reputation of the provider in terms of a ranking given by the end users [8, 3]. With such an interpretation, reputation remains merely a user perception failing to indicate the true performance levels of the provider or the service. It is also susceptible to unwanted manipulative rankings from hostile users. Reputation should be an indicator of the truthfulness and ability of the service provider to deliver the stated quality levels along with the user rating. When expressed as a vector of compliance, verity and user ranking it serves as the true indicator of the provider's trustworthiness.

4. Architecture for Verity in an SLA Environment

The architecture for a SLA-enabled web service typically consists of three components: the service provider, the end user and the service broker [6]. According to this architecture, the service provider publishes a SLA-enabled web service and sends it to the service broker for storage in a repository. An end user registers with the service broker searches and finalized the SLA with the appropriate service provider. After the provider and the end user negotiate the relevant parameters, the transaction needs to be monitored by a third party to detect any violations of service level objectives.

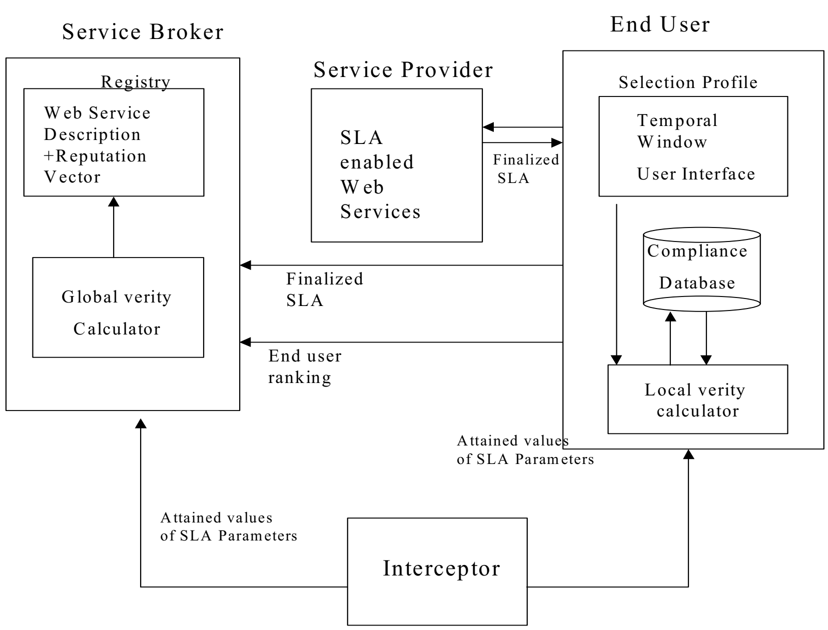

Figure 1. Architecture for verity and reputation in SLA enabled Web services

This is accomplished by the SLA monitoring model in [7] where a third party is assigned the job of verifying the values of SLA parameters in the agreement against the values obtained by probing or intercepting the client invocation. This model is restricted to taking corrective actions on behalf of the managed environment upon failing to comply with the SLA terms.

Compliance is not a new concept and has been mentioned in [4] and explored by [21] under the name of *Conforma nce*. Their design for *Conformance* consists of a contract template and a System Dictionary used to verify a contract. The customer specific data is inserted in the contract template and verified against the system measurements that were asserted while forming the contract. The contract is evaluated and the compliance results are presented as customized reports for each customer. This paper proposes extensions of the work in [8] and [3] by progressively recording the actual and achieved parameter values for each client invocation of a service and quantifying the compliance in numerical terms. This could be then coupled with verity and user-given ranking to indicate the reputation of the provider. The architecture for this proposed model is displayed in Figure 4.1.

This model enhances the functionality of the service broker and the end user with the addition of a *verity calculator* to each of them. The *interceptor* is responsible for measuring the SLA parameter values delivered at the end of each service invocation. The end user initiates this interception at the start of each service and upon completion of the service the interceptor sends these values to both the end user and the service broker for verity calculations. The end user supplies the service broker with a copy of the finalized SLA at the beginning of every service invocation from which the former obtains the agreed values of the contract.

The end user and the service broker are equipped with *local* and *global compliance databases* respectively, where records of past invocations of the service are maintained. The *local database* typically consists of lesser records than its global counterpart since it records past transactions pertaining to the end user alone in comparison to the records of all the transactions of the broker in the global database. The databases maintain these records at both the service level and the service provider level. Each record consists of *compliance, verity* and *user ranking* values against each service or service provider.

4.1Local and Global Compliance

The end user and the broker hold different views of the compliance and verity of a service and its provider since the corresponding past transaction records vary. Accordingly the end user and the service broker are equipped with individual *verity calculators*. The functionality of the two calculators is essentially the same but produce different results owing to the different inputs.

The *compliance* of each parameter in the SLA is a mean of the normalized difference in the projected and actual values of the parameter. The normalized difference could be positive or negative depending on whether the agreed upon value was greater than or lesser than the delivered value, respectively. The resulting mean when positive indicates a *positive compliance* which means the agreed values have been delivered without violations. A negative mean indicates a *negative compliance* and that the provider failed to deliver the agreed values. A mean value of zero is *ideal compliance* indicating the delivered values being exactly equal to the agreed values. Similar arguments hold true for the positive and negative values of service compliance and provider compliance.

The local calculator computes the mean over the total number of invocations of the service by the end user and the global calculator computes it over the total number of invocations of the service by all users. *Compliance* of a service is then calculated as the mean of compliance of each QoS parameter. The *compliance* of a service provider is similarly calculated as the mean of compliance values of all the services hosted by the service provider.

4.2 Local and Global Verity

Verity of a service is calculated as the degree of variance in the compliance values attained for each service invocation. Variance is a measure of spread in a population. The degree of spread in the *compliance* values of the service provider indicates how efficiently and how often the guaranteed levels are met. Thus it is a measure of trustworthiness and verity. The local *verity* is calculated over end user invocations of the service and the global *verity* encompasses the invocations by all the end users. *Verity* of a service provider is similarly the degree of variance in the *compliance* values of all the services hosted and varies in the *local* and *global databases*.

4.3 User Rating

The end user ranks the service and the service provider at the completion of each service. The local and the global calculators aggregate this ranking with the previous rankings to obtain the average ranking for the service and its provider. The local service rank is the mean of all rankings given by the end user during previous invocations of the service. This rank is used to calculate the global service rank as the mean of rankings given by each end user that invoked the service through this service broker. The local and global rankings of the provider are similarly calculated.

4.4 Temporal Window

The results of all the above expressions rely on the length of the period they are calculated on. The total number of times a service is invoked or the total number of services hosted by a provider is determined by the length of the temporal window. The end user and service broker can control this window for each calculation and the system can be directed to perform the computations over periods of days, weeks, months or years. An assessment over the past two months is an indicator of the recent performance and can be very different from a yearly performance. The temporal aspect gives the end user the flexibility to assess the provider and the associated services at different temporal granularities.

4.5 Local and Global Profiles

This architecture advocates maintaining of two profiles for each service and service provider- one local and the other global. The local profile is necessary to identify the personal preferences of the end user which typically differ from the global perspective. A service favoured by the end user might have been assessed unfavourable by the rest of the users. If the end user had only the global profile to rely on, this personal preference cannot be exercised. The global profile being unavailable under unavoidable circumstances is also an incentive to maintain a personalised profile at the user's end. This yields two different values of verity for each web service and provider.

It has been emphasized early on in this paper that the value of verity should also be taken into account for a quality driven selection of web services. To determine the value of verity that will be used in the quality driven selection, a weighted sum mechanism is employed. When the local and global profiles are both available, the user analyses the local and global values of ranking, compliance and verity and assigns weights to each of the attributes. The weight assigned to each attribute indicates its level of significance to the end user and the sum of the weights is equal to 1. The weighted sum of verity, compliance and ranking of the global and local profiles are compared and the verity value of the highest sum is used in the quality driven selection. For e.g. to emphasize verity, the weights could be assigned as Wverity=2/3 Wcompliance=1/9 Wranking=2/9 and the sum of weights =1/9=2/9+2/3=1. If the local values are Lverity=0.7 Lcompliance=-0.4 Lranking=4 and global values are Gverity=0.9 Gcompliance=-0.3 Granking=4, the weighted sums are Lweight=1.311 and Gweight=1.455. Gweight being the higher of the sums, the Gverity value 0.9 is used in the quality driven selection.

5. Formalised Calculations of Compliance, Verity, Rating and Reputation

This section formalises the computations involved in quantifying the compliance, verity and rating at both the service and service provider levels. It should be noted that the value of *n* used in the calculations is chiefly influenced by the length of the temporal window.

5.1 Compliance of a Web Service

The compliance of a service is calculated from the compliance values of all the SLA parameters. Every attribute a^i in the SLA is associated with a projected value $a^i{}_p$ as agreed while finalizing the SLA and $a^i{}_d$ is the achieved value of the parameter.
For n invocations of the service, the projected and delivered values of an attribute a^i are

$$< (a^i p_1, a^i d_1), (a^i p_2 a^i d_2), (a^i p_3, a^i d_3) ..... (a^i p_n, a^i d_n) >$$

The normalised difference in the values of $a^i p_j$ and $a^i d_j$ is represented as $\mathrm{Nd}_j^{a_i}$ and can be positive or negative. The normalised value is relative to the projected value and scales well with different values of actual and projected values.

$$\mathrm{Nd}_j^{a_i} = \frac{a^i p_j - a^i d_j}{a^i p_j}$$

For n invocations of the service the normalised difference values of an attribute a^i are represented as

$$< \mathrm{Nd}_1^{a_i}, \mathrm{Nd}_2^{a_i}, \mathrm{Nd}_3^{a_i}, ..., \mathrm{Nd}_n^{a_i} >$$

The value of each $\mathrm{Nd}_j^{a_i}$ is ideally zero which indicates that the provider is fully compliant with the agreed upon values. When the provider is not fully compliant, compliance is calculated as the mean of this normalised difference:
Local Compliance of an attribute a^i is calculated as below:

$$\mathrm{L}^{a_i}_{\mathrm{compl}} = \sum_{j=1}^{n} \frac{\mathrm{Nd}_j^{a_i}}{\mathrm{n}}$$

In the above expression, n is the number of times the service was invoked by a particular end user and $\mathrm{Nd}_j^{a_i}$ is the normalized difference of projected and delivered

values of attribute a_i when the service was invoked the j^{th} time.
Global Compliance of an attribute a^i is calculated as below:

$$G^{a_i}_{compl} = \sum_{j=1}^{n} \frac{Nd_j^{a_i}}{n}$$

In the above expression, n is the number of times the service was invoked by all the end users put together and $Nd_j^{a_i}$ is the normalized difference of projected and delivered values of attribute a_i when the service was invoked by the j^{th} end user.
Compliance of a service is the mean of compliance of all the SLA parameters. Local Compliance of a service is calculated as below:

$$WSL_{compl} = \frac{\sum_{i=1}^{n} L^{a_i}{}_{compl}}{n}$$

In the above expression, n is the number of parameters in the SLA and $L^{a_i}{}_{compl}$ is the local compliance of parameter a^i.
Global Compliance of a service is calculated as below:

$$WSG_{compl} = \frac{\sum_{i=1}^{n} G^{a_i}{}_{compl}}{n}$$

In the above expression, n is the number of parameters in the SLA and $G^{a_i}{}_{compl}$ is the global compliance of parameter a^i.

5.2 Compliance of a Service Provider

The local compliance of a service provider is the average of the compliance values of the web services invoked by the end user that were hosted by the service provider.
Local Compliance of service provider is calculated as below:

$$SPL_{compl} = \frac{\sum_{i=1}^{n} WSL^{i}{}_{compl}}{n}$$

In the above expression, $WSL^{i}{}_{compl}$ is the local compliance of a web service and n is the number of times the web service was invoked by the end user.
Global Compliance of service provider is calculated as below:

$$SPG_{compl} = \frac{\sum_{i=1}^{n} WSG^{i}{}_{compl}}{n}$$

In the above expression, $WSG^{i}{}_{compl}$ is the global compliance of a web service invoked by the i^{th} end user n is the total number of services hosted by the provider and invoked by all the end users.

The compliance value thus calculated is statistically accurate and serves as a metric for further analysis. This metric is used here to compute the verity of a service and service provider.

5.3 Verity of a Web Service

The verity of a service is the variance in compliance values of the service for each invocation. Local verity of a service is calculated as below:

$$WSL_{verity} = \frac{\sum_{i=1}^{n} (WSL_i - \mu)^2}{n}$$

In the above expression, n is the number of times the service WS was invoked by the end user and WSL^{i}_{compl} is the local compliance level when invoked the i^{th} time by end user.

Global verity of a service is calculated as below:

$$WSG_{verity} = \frac{\sum_{i=1}^{n} (WSG_i - \mu)^2}{n}$$

In the above expression, n is the number of times the service WS was invoked by all the end users and $WSG^{i}{}_{compl}$ is the global compliance level when invoked by the i^{th} end user.

5.4 Verity of a Service Provider

The verity of a service provider is the variance in the compliance levels of all the services hosted by the provider.
Local verity of a service provider is calculated as below:

$$SPL_{verity} = \frac{\sum_{i=1}^{n} (WSL^{i}{}_{compl} - \mu)^2}{n}$$

where WSL_i is the local compliance of the i^{th} web service hosted by the service provider that were invoked

by the end user n is the total number of such services.
Global verity of a service provider is calculated as below:

$$\mathrm{SPG}_{\mathrm{verity}} = \frac{\sum_{i=1}^{n} (\mathrm{WSG}^{\mathrm{i}}{}_{\mathrm{compl}} - \mu)^2}{\mathrm{n}}$$

In the above expression , $\mathrm{WSG}_{\mathrm{i}}$ is the global compliance of the ith web service hosted by the service provider and n is the total number of web services hosted by the service provider

5.7 Reputation

Reputation of a service or service provider is expressed as a vector of user ranking, compliance and verity values.
Local Reputation of a web service is expressed as below:
$< \mathrm{WSL}_{\mathrm{compl}}, \mathrm{WSL}_{\mathrm{verity}}, \mathrm{SL}_{\mathrm{rating}} >$
Global Reputation of a web service is expressed as below:
$< \mathrm{WSG}_{\mathrm{compl}}, \mathrm{WSG}_{\mathrm{verity}}, \mathrm{SG}_{\mathrm{rating}} >$
Local Reputation of a service provider is expressed as below:
$< \mathrm{SPL}_{\mathrm{compl}}\ \mathrm{SPL}_{\mathrm{verity}}, \mathrm{SPL}_{\mathrm{rating}} >$
Global Reputation of a service provider is expressed as below:
$< \mathrm{SPG}_{\mathrm{compl}}\ \mathrm{SPG}_{\mathrm{verity}}, \mathrm{SPG}_{\mathrm{rating}} >$
In the above expressions, SLrating, SGrating, SPLrating, SPGrating are local and global user ratings of a service and service provider respectively and are calculated as the mean of the user ratings over the number of times the service or service provider was ranked.

6.Conclusion

The proliferation of web services and service providers demands concrete interoperable mechanisms to identify services of superior quality. The parameters typically found in SLAs include response time, throughput, availability and reliability. While these parameters have been widely accepted to be contributors of quality, further research is required to develop methods to quantify the same.

The QoS attributes identified to date lack a metric that can indicate how effectively these attributes have been delivered. There is a need to measure the discrepancy in the agreed and delivered values of the SLA parameters. This is termed the compliance value and is further used to quantify verity- the attribute indicative of the provider's consistency in delivering the agreed levels of the QoS parameters. The end user and the service broker maintain different profiles of the verity of services and service providers. To determine the verity value used in quality driven selection, the user employs a weighted sum mechanism. Extensions are required at the service broker and the user ends to incorporate the functionality to measure verity. The projected values of the parameters are obtained from the finalized SLA and the delivered values are measured by the interceptor.

Web services minimize the need for human interaction during transactions because of which the presence of a formal expression for reputation of service providers is most desirable. Reputation has been frequently expressed as a user ranking and is therefore not a true indicator of the competence of the provider. When expressed as a vector of compliance, verity and user ranking reputation becomes a more tangible quantity. With the ability to measure a provider's verity over a period of time, end users are exposed to more reliable and trustworthy quality driven selections. We are currently working on a prototype implementation of the architecture to compute compliance and verity values and maintain global and local reputation vectors.

7. References

[1] Shuping Ran, *A Model for Web Services Discovery with QoS*, ACM SIGecom Exchange, Volume 4 Issue 1, March 2003.
[2] Gregor V. Bochmann, Brigitte Kerherve, Hanan Lutfiyya, Mohammed-Vall M. Salem and Haiwei Ye, *Introducing QoS to Electronic Commerce Applications*, Springer-Verlag Berlin Heidelberg 2001 pp. 138-147.
[3] E. Michael Maximilien, Munindar P. Singh, *Reputation and Endorsement for Web Services,* ACM SIGecom Exchanges, Volume 3 Issue 1, December 2001.
[4] Anbazhagan Mani and Arun Nagarajan, *Understanding Quality of Service for Web Services,* http://www-106.ibm.com/developerworks/library/ws-quality.html January 2002.
[5] Anna Bouch, Allan Kuchinsky, Nina Bhatti, *Quality is in the Eye of the Beholder: Meeting Users' Requirements for Internet Quality of Service*, Proceedings of the SIGCHI conference on Human factors in computing systems, April 2000.
[6] Judith M. Myerson, *Guarantee your Web Service with an SLA: Introduction, architecture and testing mechanisms,* http://www-106.ibm.com/developerworks/webservices/library/sw-sla.html, April 2002.
[7] Alexander Keller and Heiko Ludwig, *Defining and Monitoring Service Level Agreements for dynamic e-Business.* Proceedings of the 16th USENIX System Administration Conference (LISA'02), November 2002.

[8] Liangzhao Zeng, Boualem Benatallah, Marlon Dumas, Jayant Kalagnanam, Quan Z. Sheng, *Web Engineering: Quality Driven Web Service Composition,* ACM Press Proceedings of the twelfth international conference on World Wide Web**,** May 2003.
[9] Akhil Sahai, Jinsong Ouyang, Vijay Machiraju, Klaus Wurster, *BizQoS* (2001)*: Specifying and Guaranteeing Quality of Service for Web Services through Real Time Measurement and Adaptive Control*, Hewlett-Packard Labs Technical Report HPL-2001-134 http://www.hpl.hp.com/techreports/2001/HPL-2001-134.html
[10] Daniel A. Menasce, *QoS Issues in Web Services*, Internet Computing, IEEE, Volume 6 Issue 6, Nov.-Dec. 2002, pp. 72 -75.
[11] Giorgos Zacharia, Alexandros Moukas and Pattie Maes, *Collaborative Reputation Mechanisms for Electronic Marketplaces*, Decision Support Systems, 2000, Vol.29, pp. 371-388.
[12] Paul Resnick, Richard Zeckhauser, Eric Friedman, and Ko Kuwabara, *Reputation Systems*, Communications of the ACM Volume 43, Issue 12, December 2000.
[13] Marco Conti, Enrico Gregori and Fabio Panzieri , *Load Distribution among Replicated Web Servers: A QoS-based Approach,* Second Workshop on Internet Server Performance , ACM Press, Atlanta (GA), 1999.
[14] Aad van Moorsel, *Metrics for the Internet Age: Quality of Experience and Quality of Business,* Fifth International Workshop on Performability Modeling of Computer and Communication Systems, September 2001.
[15]Prasant Mohapatra, Huamin Chen, *Web Graph: A Framework for Managing and Improving performance of Dynamic Web Conten,* IEEE Journal on Selected Areas in Communications, v 20, n 7, September, 2002, pp 1414-1425.
[16] Slashdot http://slashdot.org
[17] Rohit Khare and Adam Rifkin, *Weaving a Web of Trust.* Summer 1997 issue of the World Wide Web Journal Volume 2, Number 3, pp 77-112.
[18] Upendra Sharadanand and Pattie Maes, *Social Information Filtering: Algorithms for Automating 'Word of Mouth',* Proceedings of the CHI-95 Conference, Denver , CO, ACM Press, May 1995.
[19] eBay http://www.ebay.com
[20] OnSale http://www.onsale.com
[21] Preeti Bhoj, Sharad Singhal and Sailesh Chutani, *SLA Management in Federated Environments,* HP Labs Techincal Report, December, 1998.
[22] Amir Padovitz, Shonali Krishnaswamy and Seng Wai Loke, *Towards Efficient Selection of Web Services.* Accepted for publication in the Workshop on Web Services and Agent-based Engineering (WSABE 2003), Held in conjunction with the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2003), July 2003.
[23] Mathew Merzbacher and Dan Patterson, *Measuring end user availability on the Web: Practical Experience,* Proceedings of the International Conference on Dependable Systems and Networks (DSN 2002) IEEE Press.
[24] Oscar Ardaiz, Felix Freitag and Leandro Navarro *Improving service time of Web clients using Server Redirection*, ACM SIGMETRICS Performance Evaluation Review, Volume 29 Issue 2, September 2001.