A QOS DRIVEN APPROACH FOR PROBABILITY EVALUATION OF WEB SERVICE COMPOSITIONS

Dessislava Petrova-Antonova and Aleksandar Dimov
Sofia University, Faculty of Mathematics and Informatics, Department of Software Engineering, 1164, Sofia, Bulgaria

Keywords: QoS, Composite web-services, QoS evaluation.

Abstract: Service oriented systems are becoming a major area in software engineering due to promise of low time and cost of development efforts. However, in order to enable full benefit of service oriented architecture, some technical aspects of service-level agreements, also known as Quality of Service (QoS) are important to be evaluated in formal way. Currently evaluation of web service QoS is done mainly as underlying part of a corresponding composition model and many models consider only limited number of QoS attributes. In this paper we present an approach for evaluation of composite web-services, based on analysis of how they meet their QoS, which is applicable at multiple levels of service composition as well as for wide range of QoS attributes. Our approach is based on probability theory and is illustrated with a case-study.

1 INTRODUCTION

Currently Service Oriented Architecture (SOA) based on web services is regarded as important paradigm in many areas not only in software engineering. The promise of reduced cost and time for enabling a large range of company-wide business processes shifts the research focus towards reasoning about web service compositions. On the other hand, technical aspects of service-level agreements, also known as Quality of Service (QoS) are important to be considered as they define additional constraints and non-functional requirements toward the web-services (Cardoso, 2004). Business workflows and therefore – organizations success are directly impacted by proper management of SLAs and keeping them within terms of negotiated QoS metrics.

Various classifications (Tran et al., 2009); (D'Mello et al., 2008) of QoS properties exist in the literature. However, a widely accepted standard for specification of web service quality does not exist. The first efforts in this direction have started in the context of Web Services Quality Model (WSQM) prepared by OASIS (Kim and Lee, 2005). According to WSQM, service level measurement quality is subdivided into performance measurement subfactors including response time, throughput and maximum throughput, and stability measurement sub-factors such as availability, reliability and accessibility. Unfortunately, only draft version of the WSQM is available.

QoS properties may be distinguished on the basis of their values. If the client requires the values of QoS properties to be as higher as possible, then these QoS properties can be classified as increasing or positive. Otherwise, they are classified as decreasing or negative. Examples for positive QoS properties are *availability* and *throughput*. In contrast, *response time* and *cost* are regarded as negative QoS properties.

In this context it is of paramount importance to be able to reason about QoS of web services in formal way. This will enable the activities of service selection and evaluation when more than one service satisfies the required functionality. Moreover, formal approaches will make possible dynamic composition and reconstruction of composite web-services.

The purpose of this paper is to present a systematic QoS-based approach for evaluation of web service compositions that applies for various kinds of QoS attributes and multiple levels of service composition. It is based on our previous work about QoS evaluation of web services, assisting the search process of web services (Petrova-Antonova, 2011). The approach uses probability theory to quantify the extent to which composite web services meet their quality requirements. This may serve as a basis for composite web service optimization based on quality of its constituent services. Moreover in this way, one

Petrova-Antonova D. and Dimov A..
A QOS DRIVEN APPROACH FOR PROBABILITY EVALUATION OF WEB SERVICE COMPOSITIONS.
DOI: 10.5220/0003602303210326
In *Proceedings of the 6th International Conference on Software and Database Technologies* (ACT4SOC-2011), pages 321-326
ISBN: 978-989-8425-76-8

may also be able to find bottlenecks in service compositions.

The remainder of the paper is structured as follows: Section 2 makes an overview of the related work; Section 3 presents the theoretical foundation of our approach; Section 4 describes the steps in the approach for evaluation of composite web services; Section 5 illustrates the approach with a typical case study and finally section 6 concludes the paper and states some directions for further research in the area.

2 RELATED WORK

Modeling of QoS of composite web-services has been a matter of study from various researchers in recent years. It has been mostly regarded as an underlying part of a corresponding composition model. A number of efforts that deal with the problem of evaluation of web services with respect to their composition exist in the literature (Bartalos and Bielikova, 2009); (Lin et al., 2008); (May Shan and Bishop, 2008); (Pistore et al., 2004); (Zheng and Yan, 2008), but they do not take into account formal modeling of QoS.

A number of interesting works present modeling of QoS properties within their composition framework, however, the models are restricted to several properties only. Such approaches are (Cardellini et al., 2007); (Ming et al., 2009); (Nam et al., 2009); (Zheng et al., 2004).

For example, the methodology, described in (Chang and Lee, 2010) evaluates the quality of ubiquitous web services in three dimensions: QoS (quality of services), QoC (quality of contents), and QoD (quality of devices). Further, based on Multi-Criteria Decision Making (MCDM) with preference functions, the quality factors are evaluated and priority ordering of web services is made. A heuristic approach is presented in (Alrifai et al., 2008) that achieve a close to optimal result while offering low complexity and high speed of computations. In contrast to these approaches in our work we propose more formal evaluation criteria, based on probability theory instead of preference functions.

Interesting method for service composition is proposed in (Che et al., 2009). Authors there use XML schemas as a means for service composition. The schema contains data about the conditions on which to select the service, which include more than just QoS criteria. However this approach does not count for the indeterminism that a service may not satisfy a quality requirement, i.e. they either say that it will meet the requirement in any case or will not do so at all. Instead, our approach takes into imprecision of quality attributes, by involving the mechanism of probability.

Very formal technique for service composition and composition evaluation is proposed in (Chakhar et al.). In this work a specific method for evaluation of the service upon each quality attribute is proposed. While very exhaustive, such approach may appear difficult to be applied in real practice. In our work we try to overcome this issue by employing a formal approach, applicable to any quality attribute without modifications.

Another method for QoS evaluation which takes into account imprecision in satisfaction of QoS properties use the fuzzy set theory to formalize the model (Xiong and Fan, 2008). In contrast we claim that our approach is simpler, easier for implementation and hence faster, when considering dynamic service evaluation and composition.

3 PRELIMINARIES

In this section we are presenting the background of our approach – first we define web-service compositions and then explain some basic elements of probability theory, which we use to formalize the approach for composite web service evaluation.

3.1 Composite Web Services

Software services represent abstract entities, subject to reuse and in service-oriented architecture single services may be organized into a more complicated workflow that corresponds to a given business process. Currently the most widespread methods for design of composite services are choreography and orchestration. Orchestration considers aggregation of service operations and actually represents a workflow, while choreography means aggregation of services which results in another service. In our work we make an evaluation of the composition after it has been implemented, i.e. regardless of the composition method.

For instance, several services WS_1, WS_2, etc., may be composed into a composite web-service CWS_1 as shown on Fig. 1.

As seen from the figure, some services in the composition may be complex services themselves. Further in the paper, we denote such web services as *dependent*, because their invocation and QoS properties depend on the results from execution and QoS properties of the services that they compose. When executing composite of the kind shown on fig. 1, we assume that of sub services WS_3 and WS_4, are autonomous in their execution, as they do not take input from another web services in the composition. We call such web services *independent*.

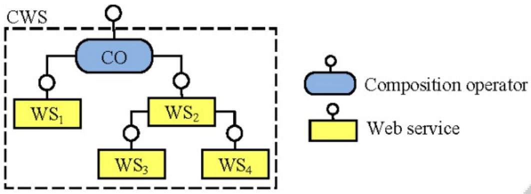

Figure 1: Composite web services.

In order to assess service qualities we use a so-called *service history log* that contains data, obtained during service execution, with concrete measures about quality parameters. History log values of dependent web services are obtained during their monitoring in testing environment, where they are isolated from web services on which they depend. History log values of independent web services are obtained during monitoring into a production environment.

3.2 Elements of Probability Theory

Generally speaking probability theory is a mathematical discipline that deals with analysis of random events. More precisely in this paper we are interested on how to estimate the likeliness that a certain event will occur. In this section we present the some aspects of the probability theory that are used in next section to leverage our approach for QoS estimation of composite web services.

Each QoS property can be represented by a discrete random variable X with several possible values x that correspond to the recorded values in the history log.

A random variable can be described by a probability mass function (PMF), which captures the probabilities of the values that the random variable can take. If x is any possible value of discrete random variable X, the PMF of x, denoted $p_X(x)$ is the probability of the event $\{X = x\}$ consisting of all outcomes that give rise to a value of X equal to x (Bertsekas and Tsitsiklis, 2000):

$$p_X = P(\{X = x\}). \quad (1)$$

For a given value x it is often necessary to compute the probability that the value of the random variable X will be at most x. This can be accomplished by a cumulative distribution function (CDF). The CDF of a discrete random variable provides the probability $P(X \leq x)$ (Bertsekas and Tsitsiklis, 2000):

$$P(X \leq x) = \sum_{k \leq x} p_x(k). \quad (2)$$

Probabilistic models usually concern several random variables. For example, in the context of web services, the QoS requirements of the web service composition consist of definitions for several QoS properties that can be seen as random variables belonging to the same experiment. The joint CDF of two independent random variables X and Y is as follows:

$$p_{X,Y}(x, y) = P(X \leq x)P(Y \leq y). \quad (3)$$

Note that for negative QoS properties probability that $\{X \geq x\}$ may be calculate as $P(X \geq x) = 1 - P(X \leq x)$.

The conditional PMF of a random variable X that is conditioned on an event A with probability $P(A)$ is defined as follows (Bertsekas and Tsitsiklis, 2000):

$$p_{X|A}(x) = P(X = x|A) = \frac{P(\{X = x\}) \cap P(A)}{P(A)}. \quad (4)$$

The conditional CDF of two random variables X and Y is as follows (Bertsekas and Tsitsiklis, 2000):

$$p_{X|Y}(x|y) = \frac{P(X \leq x, Y \leq y)}{P(Y \leq y)} = \frac{p_{X,Y}(x, y)}{p_Y(y)}. \quad (5)$$

The conditional CDF can be used for calculation of joint CDF as follows (Bertsekas and Tsitsiklis, 2000):

$$p_{X,Y}(x, y) = \mathrm{p_Y(y)p_{X|Y}(x|y)}. \quad (6)$$

4 AN APPROACH FOR EVALUATION OF COMPOSITE WEB SERVICES

This section presents a QoS driven approach for probability evaluation of composite web services. The approach consists of seven steps that are described bellow in detail.

Let web services that satisfy functional requirements of the composition form the set S:

$$S = \{WS_1, WS_2, \ldots, WS_i, \ldots, WS_n\}, i = 1 \div n \quad (7)$$

where n is the number of candidate web services.

Step 1: Identification of Independent and Dependent Web Services in the Composition. As mentioned in Section 3.1 web services in the composition can be dependent or independent according to whether a given web service uses the results from the execution of another web service in the composition or not. Therefore, the first step of the algorithm requires separation of the web services into two different sets – S' for independent web services and S'' for dependent web services.

$$S' = \{WS_1, WS_2, \dots, WS_i, \dots, WS_k\}, i = 1 \div k \quad (8)$$

$$S'' = \{WS_k, WS_{k+1}, \dots, WS_i, \dots, WS_n\}, i = k \div n \quad (9)$$

Step 2: Extraction of QoS Data. In Section 3.2 it was shown that each QoS property of web services can be represented with a discrete random variable with several possible values that correspond to the recorded values in the history log. Thus, the values of QoS properties of a particular web service form the following matrix:

$$Q_{ws_i} = \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \cdots & q_l \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1l} \\ x_{21} & x_{22} & \cdots & x_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{ml} \end{bmatrix} \end{array}, i = 1 \div n \quad (10)$$

where x_{ml} is the *m-th* value of QoS property q_l obtained from history log and m is the number of values obtained.

Step 3: Extraction of Events from Composition Requirements. The QoS requirements of the web service composition are considered as events. They form the following matrix:

$$R = \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \cdots & q_l \end{array} \\ \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1l} \\ r_{21} & r_{22} & \cdots & r_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nl} \end{bmatrix} \end{array} \quad (11)$$

where $r_{ij}=\{X \le x\}$ for negative QoS properties and $r_{ij}=\{X \ge x\}$ for positive QoS properties.

Step 4: Calculation of PMF for each Unique Value of each QoS Property for all Web Service in the Composition. The probabilities of the values that given QoS property takes are calculated according to equation (1).

Step 5: Calculation of CDF for each QoS Property of Independent Web Services in the Composition. The CDFs calculated for each QoS property of all independent web services form the following matrix:

$$P' = \begin{array}{c} \\ WS'_1 \\ WS'_2 \\ \vdots \\ WS'_k \end{array} \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \dots & q_l \end{array} \\ \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1l} \\ P_{21} & P_{22} & \cdots & P_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ P_{k1} & P_{k2} & \cdots & P_{kl} \end{bmatrix} \end{array} \quad (12)$$

CDFs $P_{ij} = P(r_{ij})$ for that case are calculated according to equation 2.

Step 6: Calculation of Conditional the CDF for Each QoS Property of Dependent Web Services in the Composition. The conditional CDFs calculated for each QoS property of all dependent web services form the following matrix:

$$P''_{X|Y_1 \div Y_z} = \begin{array}{c} \\ WS''_1 \\ WS''_2 \\ \vdots \\ WS''_{n-k} \end{array} \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \dots & q_l \end{array} \\ \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1l} \\ P_{21} & P_{22} & \cdots & P_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n-k1} & P_{n-k2} & \cdots & P_{n-kl} \end{bmatrix} \end{array} \quad (13)$$

where z is the number of web services, from which a given web service S_i takes input. Here, the conditional CDFs P_{ij} are calculated based on QoS data obtained during execution of web service in testing environment, according to equation (5). Note that in this way we consider that dependent web-services are executed given that all other services they invoke have met their quality requirements.

Step 7: Calculation of Joint CDF for QoS Properties of the Web Service Composition. Finally, we calculate the probability the QoS properties of the composition to satisfy preliminary defined requirements using equation 6. The result is as follows:

$$Q_{CWS} = \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \cdots & q_l \end{array} \\ [\begin{array}{cccc} p_1 & p_2 & \cdots & p_l \end{array}] \end{array} \quad (14)$$

Pseudo code of the algorithm that may serve as a basis for implementation of the proposed approach is presented on Listing 1.

Listing 1. Pseudo code of the QoS driven algorithm for probability evaluation of web service composition

```
FUNCTION Evaluate (log, requirements)
FOR EACH web service in S DO
   Extract data from history log
   IF web service is independent THEN
      Insert web service in the set S'
   ELSE
      Insert web service in the set S''
   END IF
   Create matrix Qws
END FOR
FOR EACH QoS requirement of the composi-
tion
   Extract an event
   Insert an event into matrix R
END FOR
FOR EACH web service in S DO
   FOR EACH QoS property DO
      FOR EACH unique value of QoS prop-
erty DO
            Calculate PMF
      END FOR
      IF web service is independent THEN
         Calculate CDF
      ELSE
         Calculate conditional CDF
      END IF
   END FOR
END FOR
FOR EACH QoS of the composition
   Calculate joint CDF
END FOR
END FUNCTION
```

5 CASE STUDY

This section shows how to apply our approach with a case-study that represents a *Travel Booking* sample, taken from (Travel Booking Sample, 2011). A business process of the sample invokes four web services as shown in Table 1.

Table 1: Web services of the Travel Booking business process.

Web service name	*Symbol*	*Type*
checkCreditCard	WS_1	Independent
checkFlightReservation	WS_2	Dependent
checkHotelReservation	WS_3	Dependent
checkCarReservation	WS_4	Dependent

The composition diagram, corresponding to the business process is shown on Fig. 2. The QoS properties that will be used in the example are Successful execution rate *(q1)*, Reputation *(q2)*, Availability *(q3)* and Response time *(q4)*. Here, the execution rate is successful when its value is 1.

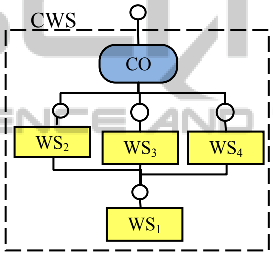

Figure 2: Travel booking process.

According to ***Step 1*** of the proposed approach, the web services in the composition form two sets depending on whether the web service is independent or not. These sets are defined as follows:

$$S' = \{WS_1\} \tag{15}$$

$$S'' = \{WS_2, WS_3, WS_4\} \tag{16}$$

On the ***Step 2*** we have to extract data from history log in order to define the QoS matrices for each web service:

$$Q_{WS_1} = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} 1 & 0.90 & 0.80 & 5.00 \\ 1 & 0.80 & 0.95 & 4.10 \\ 1 & 0.75 & 0.90 & 4.50 \\ 1 & 0.80 & 0.85 & 4.50 \\ 1 & 0.80 & 0.85 & 4.80 \\ 1 & 0.95 & 0.75 & 4.00 \\ 1 & 0.80 & 0.80 & 4.60 \end{bmatrix} \end{array} \quad Q_{WS_2} = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} 1 & 0.60 & 0.75 & 5.00 \\ 1 & 0.80 & 0.95 & 5.10 \\ 1 & 0.75 & 0.91 & 5.20 \\ 1 & 0.80 & 0.85 & 4.80 \\ 1 & 0.80 & 0.95 & 5.00 \\ 0 & 0.75 & 0.92 & 8.00 \\ 1 & 0.80 & 0.80 & 4.60 \end{bmatrix} \end{array}$$

$$Q_{WS_3} = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} 1 & 0.90 & 0.80 & 5.00 \\ 1 & 0.80 & 0.95 & 5.00 \\ 1 & 0.85 & 0.90 & 5.20 \\ 1 & 0.85 & 0.85 & 4.40 \\ 1 & 0.88 & 0.85 & 4.50 \\ 1 & 0.95 & 0.90 & 4.00 \\ 1 & 0.92 & 0.90 & 4.60 \end{bmatrix} \end{array} \quad Q_{WS_4} = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} 1 & 0.90 & 0.85 & 5.00 \\ 1 & 0.80 & 0.75 & 5.00 \\ 1 & 0.70 & 0.70 & 5.20 \\ 1 & 0.80 & 0.85 & 4.50 \\ 1 & 0.80 & 0.85 & 4.60 \\ 1 & 0.75 & 0.80 & 4.80 \\ 0 & 0.80 & 0.80 & 8.60 \end{bmatrix} \end{array} \tag{17}$$

On the ***Step 3*** the composition requirements are extracted. The events that are obtained from the requirements form the following matrix:

$$R = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} \{q_1 = 1\} & \{q_2 \geq 0.75\} & \{q_3 \geq 0.80\} & \{q_4 \leq 5.0\} \\ \{q_1 = 1\} & \{q_2 \geq 0.65\} & \{q_3 \geq 0.85\} & \{q_4 \leq 6.0\} \\ \{q_1 = 1\} & \{q_2 \geq 0.85\} & \{q_3 \geq 0.85\} & \{q_4 \leq 5.0\} \\ \{q_1 = 1\} & \{q_2 \geq 0.75\} & \{q_3 \geq 0.75\} & \{q_4 \leq 7.0\} \end{bmatrix} \end{array} \tag{18}$$

According to ***Step 4***, the probabilities of the values that the QoS property of each web service takes are calculated. They are used to compute the CDF of each QoS property of independent web services on the ***Step 5*** and the conditional CDF of each QoS property of dependent web services on the ***Step 6***. The result is as follows:

$$P' = \begin{array}{cc} & \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ WS_1 & \begin{bmatrix} 1 & 1 & 0.857 & 1 \end{bmatrix} \end{array} \tag{19}$$

$$P'' = \begin{array}{cc} & \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{matrix} WS_2 \\ WS_3 \\ WS_4 \end{matrix} & \begin{bmatrix} 0.857 & 0.857 & 0.857 & 0.857 \\ 1 & 0.857 & 0.857 & 0.857 \\ 0.857 & 0.857 & 1 & 0.857 \end{bmatrix} \end{array} \tag{20}$$

Finally, on ***Step 7*** the joint CDF for QoS properties of the web service composition is calculated.

$$Q_{WSC} = \begin{array}{c} \begin{matrix} q_1 & q_2 & q_3 & q_4 \end{matrix} \\ \begin{bmatrix} 0.735 & 0.63 & 0.463 & 0.63 \end{bmatrix} \end{array} \tag{21}$$

Given the relatively small number of input data, the fact that we get probability values less than 0.80 may be considered as good result. Given the requirements for the composite web-service, the results may be optimized by selecting a better candidate web services.

6 CONCLUSIONS AND FUTURE WORK

There is a growing need for establishment of sound, industry-wide techniques and methodologies for service composition, not only in design time, but also in run-time. The later compositions are usually called dynamic web-services. For this purpose an important assessment criteria for candidate services is fulfillment of their quality attributes. This paper proposes an approach for evaluation of composite web services, based on probability analysis of how do they meet their QoS requirements. This approach is applicable for evaluation of any set of QoS by just calculating the probability if a given QoS would be satisfied or not, based on service execution history log.

Another benefit of our approach is that it uses probabilities for evaluation of QoS properties and many of them like reliability and availability are actually measured with the same metrics.

Main directions for future research include enabling a framework for dynamic service composition, based on proposed approach. As short-term future work, we plan to implement the proposed algorithm and experiment with various industrial examples of web service compositions. Another direction for further research is to enrich our approach with ability to reason about timing and sequence of service executions.

ACKNOWLEDGEMENTS

The work presented in this paper was partially supported by grants from the National Science Fund, in Bulgaria under the MU-01-143 (ADEESS) project and the SISTER project, funded by the EC in FP7-SP4 Capacities via agreement no.: 205030.

REFERENCES

Alrifai, M., Risse, T., Dolog, P., Nejdl, W.: A Scalable Approach for QoS-based Web Service Selection. In *Proceedings of ICSOC Workshops* (2008) 190-199.

Bartalos, P., Bielikova, M.: Fast and Scalable Semantic Web Service Composition Approach Considering Complex Pre/Postconditions. In: *International Workshop on Web Service Composition and Adaptation*, (2009) 414–421.

Bertsekas, D., Tsitsiklis J.: Introduction to Probability. Cambridge, Massachusetts, USA (2000).

Cardellini, V., Casalicchio, E., Grassi, V., Lo Presti, F.: Flow-Based Service Selection for Web Service Composition Supporting Multiple QoS Classes. In: *IEEE International Conference on Web Services* (2007) 743-750.

Cardoso, J., Miller, J., Sheth, A., Arnold, J.: Modeling Quality of Service for Workflows and Web Service Processes. *Web Semantics Journal: Science, Services and Agents on the World Wide Web*, Vol. 1, Issue 3, Elsevier Inc, MA, USA (2004)281-308.

Chakhar, S., Youcef, S., Mousseau, V., Mokdad, L., Haddad, S.: Multicriteria Evaluation-Based Conceptual Framework for Composite Web Service Selection. Available at http://www.lipn.univ-paris13.fr/~youcef/BookQoS.

Chang H., Lee K.: A Quality-Driven Web Service Composition Methodology for Ubiquitous Services. In: *Jornal of Information Science and Engineering* 26(6) (2010) 1957-1971.

Che, H., Li, Y., Oberweis, A., Stucky, W.: Web Service Composition Based on XML Nets. In: *42st Hawaii International Conference on Systems* (2009) 1-10.

D'Mello, D. et al.: A QoS Broker Based Architecture for Dynamic Web Service Selection. Proceeding of the *Second Asia International Conference on Modelling & Simulation*. Kuala Lumpur, Malaysia, (2008) 101-106.

Farhan, H. K., Javed, M. Y., Saba, B. and Sikandar, H. K.,: QoS Based Dynamic Web Services Composition & Execution. In: *International Journal of Computer Science and Information Security*, Vol. 7, Issue 2, USA (2010) 147-152

Kim, E., Lee, Y.: OASIS: Quality Model for Web Services. 2005, http://www.oasis-open.org/committees/download.php/15910/WSQM-ver-2.0.doc.

Lin, N., Kuter, U., Sirin, E.: Web service composition with user preferences. In: *5th European semantic web conference on the semantic web: research and applications,* Lecture Notes in Computer Science, Vol. 5021, Spain (2008) 629-643.

May Chan K. S., Bishop J.: A Self-healing Cycle for Web Service Composition. In: *17th International World Wide Web Conference* (2008).

Ming, Q., Khendek, F., Serhani, A., Dsouli, R., Roch, G.: An Architecture for Automatic QoS Adaptation for Composite Web Services. In: *International Journal of Web Services Practices*, Vol. 4, No.1 (2009) 18-27.

Nam, W., Kil, H., Lee, J.: QoS-Driven Web Service Composition Using Learning-Based Depth First Search. In: *IEEE Conference on Commerce and Enterprise Computing* (2009) 507-510.

Pistore, M., Barbon, F., Bertoli, P., Shaparau, D., Traverso, P.: Planning and Monitoring Web Service Composition. *Lecture Notes in Computer Science Vol. 3192* (2004), 106-115

Petrova-Antonova D.: A QoS Aware Approach for Web Service Selection Based on Probability Evaluation. *IADIS International Conference Informatics* 2011, Rome, Italy, 20 - 22 July (2011) (to appear).

Tran, V. et al.: A new QoS ontology and its QoS-based ranking algorithm for Web services. In *Simulation Modelling Practice and Theory*. Vol. 17, No. 8, (2009) 1378-1398.

Zheng, X., Yan, Y.: An Efficient Syntactic Web Service Composition Algorithm Based on the Planning Graph Model. In: *8th IEEE International Conference on Web Services* (2008) 691-699.

Xiong P., Fan, Y.: QoS-aware Web Service Selection by a Synthetic Weight. In: *IEEE Transactions on Systems, Man, and Cybernetics,* Part A, Vol. 38, Issue 4, (2008) 888-895.

Zeng, L., Benatallah, B., Ngu, A., Dumas, M., Kalagnanam, J. T., Chang, H.: In: QoS-aware Middleware for Web Services Composition. *IEEE Transactions on Software Engineering*, Vol. 30 Issue 5 (2004) 311-327.

Travel Booking Sample, (IBM WebSphere), last visited 05 April 2011. http://publib.boulder.ibm.com/infocenter/adiehelp/v5r1m1/index.jsp?topic=/com.ibm.etools.ctc.bpel.doc/samples/travelbooking/travelBooking.html.