Extending OWL for QoS-based Web Service Description and Discovery

Kyriakos Kritikos

University of Crete, Heraklion, Greece,
kritikos@csd.uoc.gr

1 Main Theme of Thesis

Web Services (WSs) are modular, self-describing, loosely-coupled, platform and programming language-agnostic software applications that can be advertised, located and used across the Internet. They are viewed as one of the promising technologies that could help business entities to automate their operations on the web on a large scale by automatic discovery and consumption of services. Based on the above reasons, the WS paradigm is being adopted by many companies and individuals and many WSs are being deployed and running.

However, as all of these WSs are advertised in a UDDI-based repository, an unavoidable fact as UDDI is a de-facto standard, the problem of discovering them based on a requester's functional needs becomes crucial. UDDI uses a syntax-based approach for WS description leading to purely syntactic discovery efforts returning imprecise and inaccurate results. **OWL-S** [OWL-S Coalition 2003] and similar joint **Semantic Web** and WS efforts solve the problem of syntactic WS description by using ontologies for describing WSs. Ontologies provide meaning to concepts and relationships between them, leading to semantic **WS Discovery** algorithms, which provide more precise and accurate results.

But even if all the advertised WSs satisfying a requester's functional needs are returned, many results may be produced. So a non-functional concept is needed that will differentiate between the functionally equivalent WS advertisements. This concept is **quality of service** (**QoS**). QoS is closely related with the performance of a WS as well as with other features of a WS that bear on its ability to satisfy stated or implied needs. Therefore it has a substantial impact on users' expectations from a service. Thus WS descriptions must be enhanced with QoS descriptions. Additionally, WS discovery algorithms should perform QoS-based matchmaking and selection in order to produce fewer ranked results.

Unfortunately, all the current research efforts fail in correctly describing QoS for WSs. **Semantics** seems to be missing from the QoS description of a WS leading to purely syntactic QoS-based WS matchmaking and selection algorithms. But even if semantics is introduced, QoS description is not rich enough and not quite extensible. So the **main issue** of this PhD thesis is the **rich, extensible, and semantic description of QoS for WSs**. Additionally, **new** QoS-based WS **matchmaking and selection algorithms** must be **devised** or the **current best** should be **extended** in order to take advantage of this enhanced semantic QoS description.

2 Motivation for Research

After reviewing several definitions for QoS for WSs, we consider **QoS** for a WS as "*a set of non-functional characteristics/attributes that may impact the quality of the service offered by the WS*". If a WS is advertised to have certain values (or range of values) in these QoS attributes, then it is said that this WS **conforms** to a certain **QoS level**. In this section we explain the reasons for incorporating QoS in WS description.

According to [Cardoso et. al. 2004], several researchers have identified Web Processes (WPs) as the computing model that enables a standard method of building Web-services applications and processes to connect and exchange information over the Web. For organizations, the ability to characterize WPs based on QoS has **four distinct advantages**. **First**, it allows organizations to translate their vision into their business processes more efficiently, since WPs can be designed according to QoS metrics. For e-commerce processes it is important to know the QoS an application will exhibit before making the service available to customers. **Second**, it allows for the selection and execution of WPs based on their QoS, to better fulfil customer expectations and requirements. **Third**, it makes possible the monitoring of WPs based on QoS to assure compliance both with initial QoS requirements and targeted objectives. QoS monitoring allows adaptation strategies to be triggered when undesired metrics are identified or when threshold values are reached. **Fourth**, it allows for the evaluation of alternative strategies when adaptation becomes necessary. **It is essential** that the services rendered follow customer specifications to meet their expectations and ensure satisfaction. *Customer expectations and satisfaction can be translated into the quality of service rendered.* Organizations have realized that quality of service management is an important factor in their operations.

As WPs are composed or single WSs, all the above advantages of QoS management of WPs also apply to WSs. So WSs can be designed and implemented according to QoS metrics (properties). They can also be discovered and selected based on their QoS capabilities. In addition, they can be monitored in order to reassure the promised QoS levels to the customers. Moreover, monitoring of QoS for WSs can trigger adaptation strategies when undesired metrics are identified, threshold values are reached, network or software or hardware errors happen. Now, we will closely examine the advantages of QoS description (management) in other non-basic activities/functions of the Service Oriented Architecture.

After the process of **WS Selection**, the requester chooses the best WS from an ordered WS advertisement list. However, even if WS clients find the appropriate WS, they are not confident that the WS's described QoS levels will actually be delivered during **WS execution**. For this reason, the WS client and provider **enter a multi-step negotiation phase**, where they try to agree on a trusted third-party entity monitoring QoS levels delivered, on the penalties that will be imposed when one of the two main parties does not keep up with its promises, and on the validity period of the promises. The **result** of this negotiation phase is a **contract** or a **Service Level Agreement** (**SLA**) document that will give confidence and trust to the entities providing and consuming the service and will lead and guide the process of WS Execution. If agreement is not met, the negotiation is stopped and the WS client contacts the next WS from the returned list of the WS Selection phase.

When **composing** a WS, **component services** are associated to the individual tasks of the composite WS and are invoked during each execution of the WS. However, the number of services providing a given functionality may be large and constantly changing and some of these services will not always be available due to network problems, software evolution and repair, and hardware problems. One solution to this problem is given at **design time** by QoS-based WS discovery. Another solution is the **runtime selection** of component services, during the execution of a composite WS, based on quality criteria (i.e. constraints and preferences) and following a local [Benatallah et. al. 2002] or a global [Zeng et. al. 2003] selection strategy. In the latter case, **service selection** is formulated as an **optimization problem** and linear programming is used to compute optimal execution plans for composite services.

3 Review of Related Work

In this section, the current research approaches for QoS-based WS Description and Discovery are described and their deficiencies are analyzed.

The **Web Service Description Language** (**WSDL**) and **UDDI** WS *standards* are **syntactical** approaches that do not express the QoS aspect/part of WS Description. While **OWL-S** is a *standard* **semantic** approach for WS Description, it does not describe QoS offers or demands as it only contains an attribute used for rating a WS.

[**Tosic et. al. 2002**] argue that for the specification of constraints for QoS metrics/attributes, five ontologies must be developed from which the most important (the top one) is the metrics ontology. They describe the structure and involved elements in four out of the five ontologies but they did not develop any ontology. In addition, the requirements specified are incomplete as each from the four aspects of QoS description needs further analysis.

In [**Shuping Ran 2003**], an extension to UDDI is proposed that represents description of QoS information about a particular WS. However, there is no actual description of the contents of this extension apart from its structure. Moreover, it relies on the UDDI (model), so it can be used only for syntactic matchmaking of offers and demands.

In [**Maximilien and Singh 2002**], an architecture and a conceptual model of WS reputation (QoS) (which encloses a QoS attributes model) are presented. However, the reputation of a WS is calculated and not it's QoS. In addition, not only concepts like QoS constraints and QoS offers and demands are not modeled but also the QoS metrics model is not rich enough.

Work described in [**Tosic et. al. 2003**], which presents the **Web Service Offerings Language** (**WSOL**), proposes that a WS must offer different classes of service in order to satisfy a greater amount and type of customers and in order to deal successfully with situations where there is a variation in QoS due to network problems or mobility reasons. This work comes with the following shortcomings: (a) no separation and integration of constraint dimensions; (b) no specification of a QoS demand; (c) the metrics ontologies are not yet developed.

The research effort described in [**Tian et. al. 2003**] analyzes what must be enclosed into the QoS information for a WS request or advertisement with the help of a QoS ontology. However, not only there is not a complete and accurate description of QoS constraints but also metrics ontologies are not developed but just referenced.

In [**Zhou et. al. 2004**], DAML-S WS description language is extended to include a QoS specification ontology. In addition, a novel QoS matchmaking algorithm is proposed, which is based on the concept of **QoS profile compatibility**. The deficiencies of this research effort are the following: (a) The metrics model is not rich enough; (b) QoS metrics have the set $\mathbb{N}^+$ as their range; (c) DL reasoners are slow and do not support the most complex mathematical expressions.

The research effort described in [**Martín-Díaz et. al. 2003**] uses a symmetric QoS model expressing mathematical constraints for QoS metrics. However, **semantics is missing** leading to syntactic matchmaking and selection algorithms. Before matchmaking, a QoS specification is transformed to a **Constraint Satisfaction Problem** (**CSP**) [Hentenryck and Saraswat 1996] which is checked for **consistency** (if it has any solution). Matchmaking is performed according to the concept of **conformance** (if every solution of offer is a solution of demand). Concerning WS Selection, the (QoS) score of a WS advertisement is expressed as a **Constraint Satisfaction Optimization Problem** (**CSOP**) [Hentenryck and Saraswat 1996] where from all solutions to the CSP of an offer we try to find the one that minimizes the weighted sum of the weight of each metric multiplied with its utility assessment value. Unfortunately, CS(O)Ps can have non-polynomial solutions when there are non-linear expression at QoS constraints.

4 Future Work of the Thesis

Based on the previously analyzed research work, we propose the following steps that must be taken to address the issues of the thesis; the fulfilment of which will lead to correct, efficient and accurate QoS-based WS Description and Discovery processes. Some of them have already been performed while the other will be dealt with in the future. These steps are:

Requirements for QoS-based WS Description: This step has already been taken. We have come up with the following requirements:

- Extensible and formal semantic QoS model
- Standards compliance
- Syntactical separation of QoS-based and functional parts of service specification
- Both requester and provider QoS specification
- Refinement of QoS specifications (extensibility, reusability)

- Fine-grained QoS specification (QoS specs for the whole WS and its parts)
- Extensible and formal QoS attribute/metric model which must at least specify: **(a)** The value set of the attribute; **(b)** The domain of knowledge of the attribute; **(c)** The relationship of the attribute with other attributes; **(d)** The association of the attribute with a unit, a measured property and a measurement function; **(e)** A functional description of how a QoS attribute of a complex WS can be derived from the corresponding QoS attribute of the individual WSs that constitute the complex one.
- Classes of service (an advertisement should present many offers)

QoS-based WS Description : Based on the above requirements, a QoS-based WS ontology has been developed with the name **OWL-Q**. This ontology extends **OWL-S** (standard) WS Description ontology and is carefully separated into many facets, each capturing one aspect of QoS WS (metric) description except from the basic one that associates OWL-S elements with OWL-Q elements. This ontology satisfies the above requirements and enables the semantic matchmaking of QoS advertisements and offers.

Semantic QoS metric matchmaking: Based on the OWL-Q ontology, we have developed an algorithm that semantically matches QoS metrics of demands and offers. This algorithm can be used as a building block for the QoS-based WS matchmaking and selection algorithms. For matchmaking simple metrics, this algorithm compares the type and measurement directive of the metrics, while for composite metrics it also compares the metrics' measurement functions.

Develop a new or extend an old QoS-based WS matchmaking algorithm: We have extended the matchmaking algorithm proposed in [Martín-Díaz et. al. 2003] in order to incorporate the semantic QoS metric matchmaking algorithm. To be more specific, when transforming QoS offers and demand to CSPs, we are careful to assign same metrics to same CSP variables and to perform unit transformation of the same metrics.

Develop a new or extend an old QoS-based WS selection algorithm: We have extended the selection algorithm proposed in [Martín-Díaz et. al. 2003] in order to incorporate the semantic QoS metric matchmaking algorithm. To be more specific, when transforming QoS offers to CSOPs, we are careful to assign same metrics to same CSP variables and to perform unit transformation of the same metrics. We have also changed the scoring function: now this function is the partial sum of the minimum and maximum assessment sums. That is we solve two CSOPs for the same offer and then we perform a partial sum of their results.

Implement these matchmaking and selection algorithms: This is a step under development. The implementation uses an OWL inference engine for the semantic QoS metric matching and the ECLiPSe [Novello and Schimpf 1999] engine for solving CS(O)Ps.

Formal evaluation of the above algorithms: The above three algorithms should be formally evaluated in order to prove that they are efficient, quick, accurate and precise.

Extend the ontology and the algorithms: After performing all the above steps, OWL-Q ontology should be extended to include other non-functional descriptions of WSs (mainly contextual ones) and its design must be finalized. In addition, the QoS-based matchmaking algorithm must be extended in order to distinguish between soft and hard non-functional constraints.

Tools: GUIs and other utilities should be constructed that will help the user in describing and discovering WSs.

References

BENATALLAH, B., DUMAS, M., SHENG, Q. Z., AND NGU, A. 2002. Declarative Composition and Peer-to-Peer Provisioning of Dynamic Web Services. In *Proc. of ICDE'02*, IEEE Computer Society, pages 297–308, San Jose.

CARDOSO, J., SHETH, A., MILLER, J., ARNOLD, J., AND KOCHUT, K. 2004. Quality of Service for Workflows and Web Service Processes. Journal of Web Semantic (to appear in 2004).

HENTENRYCK, P., AND SARASWAT, V. 1996. Strategic directions in constraint programming. *ACM Computing Surveys*, 28(4), December 1996.

MARTÍN-DÍAZ, O., RUIZ-CORTÉS, A., BENAVIDES, D., DURAN, A., AND TORO, M. 2003. Automating the procurement of Web Services. In *Proc. of the International Conference on Service Oriented Computing (ICSOC'03), Springer LNCS 2910*, pages 91–103, Berlin, Germany.

MAXIMILIEN, E.M., AND SINGH, M. 2002. Conceptual model of web service reputation. SIGMOD Record, ACM Special Interest Group on Management of Data (2002) 36–41.

NOVELLO, S., AND SCHIMPF, J. 1999. ECLiPSe Embedding and Interfacing Manual.

OWL-S COALITION (2003): OWL-S 1.0 Release. `http://www.daml.org/services/owl-s/1.0/`.

RAN, SHUPING. 2003. A Model for Web Services Discovery with QoS. A*CM SIGecom Exchanges*, 4(1):1–10.

TIAN, M., GRAMM, A., NABULSI, M., RITTER, H., SCHILLER, J., AND T. VOIGT, T. 2003. QoS Integration in Web Services. Gesellschaft fur Informatik DWS 2003, Doktorandenworkshop Technologien und Anwendungen von XML (Ph.D. students workshop Technologies and Applications of XML), Berlin, Germany, Oct. 2003.

TOSIC, V., ESFANDIARI, B., PAGUREK, B., PATEL, K. 2002. On Requirements for Ontologies in Management of Web Services. In *Proceedings of the Workshop on Web Services, e-Business, and the Semantic Web - WES* (at CAiSE'02), Bussler, C. et al. (eds.), Toronto, Canada, May 2002. Springer-Verlag, Lecture Notes in Computer Science (LNCS), No. 2512, pp.237–247.

TOSIC, V., PAGUREK, B., AND PATEL, K. 2003. WSOL — A Language for the Formal Specification of Classes of Service for Web Services. In the *Int. Conf. on Web Services (ICWS03).* Las Vegas, USA, June 23–26, 2003, CSREA Press, pp. 375–381.

ZENG, L., BENATALLAH, B., DUMAS, M., KALAGNANAM, J., SHENG, Q. 2003. Quality driven web services composition. In: *WWW 2003*: 411–421.

ZHOU, C., CHIA, L-T., AND LEE, B-S. 2004. DAML-QoS Ontology for Web Services. In: *ICWS 2004*: 472–479.