A Web Service Recommender System Using User Ontology (CSIT 2009)*

Sajjad Haj Shahvalady
Department of computer engineering,
Islamic Azad University of Qazvin,
Qazvin, Iran
Shahvalady@qazviniau.ac.ir

Mehran Mohsenzadeh
Department of computer engineering,
Islamic Azad University, Sciences & Research Center, Tehran, Iran
mohsenzade@sr.iau.ac.ir

Azadeh Ghari Neiat
Department of computer engineering,
Islamic Azad University North Tehran Branch,
Tehran, Iran
a_gharineiat@ieee.org

Fariborz Mahmoodi
Department of computer engineering,
Islamic Azad University of Qazvin,
Qazvin, Iran
fzmahmoodi@gmil.com

ABSTRACT
As the number of available Web Service (WS) providers expands, redundancy will become prevalent with many service providers offering the same or similar service. In this paper, we propose a Web service recommendation framework which in this paper is referred to as recommender that analyzes the created generalized WSDL based on the functional and non-functional requirements (i.e. QoS) and then recommends it to selected Web service providers to increase their retrieval probability in the related queries. The proposed framework gives Web services consumers and providers some confidence about the quality of service of the discovered and published Web services.

Keywords
Semantic Web service discovery, Web service recommendation, QoS, recommender.

1. INTRODUCTION

Web service technology is one of the most promising technologies in distributed computing area. Lately, developments in Web service discovery approaches mainly focus on the concept of the QoS (quality of service). With the sharp increase of the service number, many web services are in fact providing the same functions which make QoS a very important issue in distinguishing and ranking services with similar functionality.

Previous researches have discussed about WS QoS models, definition, classification and QoS modeling and so on. Papers [4, 5] propose a QoS model. Service qualities are classified into four categories: user's point of view, system level view, service level view, and business level view. Then the researchers select 5 service attributes to evaluate the web service: execution price, execution duration, reputation, successful execution rate and availability. Paper [7] introduces a method to extend the Web Service Repository Builder (WSRB) architecture by offering a quality-driven discovery of web services and uses a combination of web service attributes as constraints while searching for relevant web services. Paper [8] proposes a higher level framework for WS performance analysis and a recommendation based on the performance experienced by the client. The framework is divided into an ongoing analysis process and an on demand recommendation.

Other approaches focus on improving the selection process of web services. Paper [9, 10] develops a middleware for enhancing web service composition for monitoring QoS metrics. However, many of the researches mainly focus on the QoS model establishment and ranking the services in a static way. These studies may be more reasonable if the ranking process can be extended to a dynamic way.

QoS is a set of non-functional attributes that may impact the quality of the service offered by the WS. Each QoS attribute is measured by one or more QoS metrics, which specify all appropriate measurement details. A QoS-based WS specification is materialized as a set of constraints on a certain set of QoS metrics.

These constraints restrict the values the metrics can take. Actually, the current QoS-based WS specifications efforts only differ in the expressiveness of these constraints. However, these efforts fail in QoS metrics modeling as their model is syntactic, poor and not extensible. This has the effect on QoS-based WS discovery algorithms, which also present serious shortcomings, of producing irrelevant or incomplete results [6].

The purpose of this paper is to consider the dynamic factors in service running and to adjust the ranking of service based on the user preference.

Our previous work [1, 2] demonstrated a framework for semantic Web service discovery based on the technology of agent and semantic Web services using user ontology. We demonstrated that generalized WSDL can be recommended to selected Web service providers because if Web service

providers describe themselves more complete their retrieval probability in the related queries will increase. In this framework we use the functional and non-functional requirements to evaluate the created generalized WSDL.

The rest of this paper is organized as follows: Section 2 proposes a framework for recommender, and then gives an algorithm for certifying generalized WSDL based on functional and non-functional requirements. Section 3 outlines our plans for future works and concludes.

2. PROPOSED FRAMEWORK FOR RECOMMENDER

Based on the prior Web service discovery framework, a recommender framework is proposed, which aims at providing the service requester with appropriate services in terms of user interest profile given by the ontology manager in [2].

The basic consideration behind the idea of recommender is that if Web service providers describe themselves more complete their retrieval probability will increase. Thus, it would be of great help for the provider if there is a computer system which may provide him or her with a recommendation of a generalized WSDL.

Fig 2 explains the proposed architecture of recommender in detail while showing its relation with outside world (Fig 3), too.

As depicted by Fig 2, the proposed recommender framework contains two parts, functionality evaluation and QoS checker.

We assume here that the extraction component is an independent component to extract generalized WSDL functionalities for web service quality evaluation by comparator. The functionalities comparator receives the extracted functionalities of generalized WSDL and compare with the functionalities of this group that have been stored in functional registry using FMCA (Functional Metrics Certification Algorithm) algorithm to evaluate generalized WSDL functionalities quality. The generalized WSDL functionalities quality is evaluated and a response is sent to decision engine.

QoS is an important attribute in selecting WS providers, when there is more than one WS provider offering the same or similar service. Although there are many different definitions of QoS of WS [3, 13-15], performance is one of the important QoS attributes identified by all researchers [11].

The QoS checker receives the generalized WSDL and verifies QoS of the generalized WSDL using QSMCA (QoS Metrics Certification Algorithm) algorithm with QoS registry, and then sends the result to decision engine.

After evaluating the generalized WSDL functionalities quality and QoS, decision engine using decision algorithm evaluate these results. Then a response is sent to CU by decision engine whether if the generalized WSDL has been accepted or declined for recommending to selected Web service providers. Finally, CU recommends accepted WSDL to selected Web service providers; CU is informed of the results of recommendation.

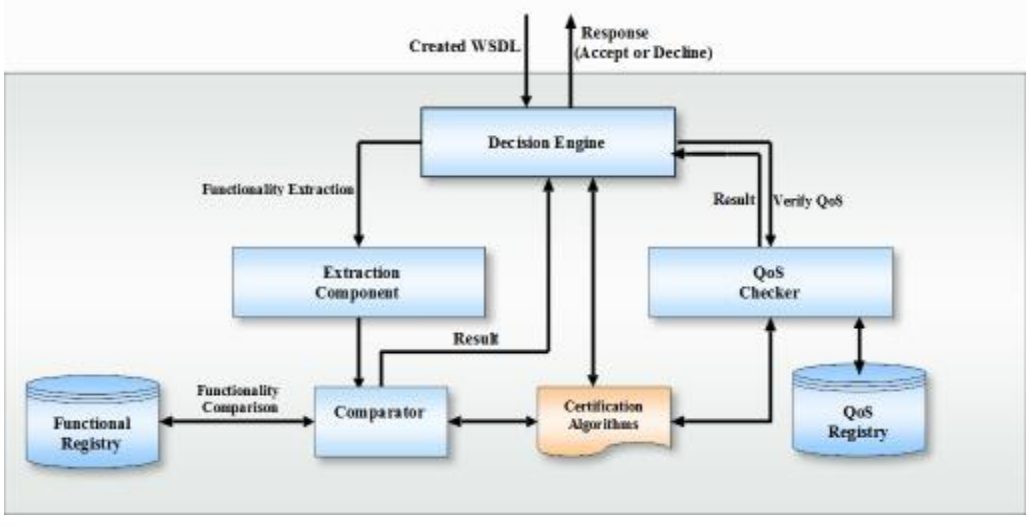

Fig 1. Proposed framework of recommender

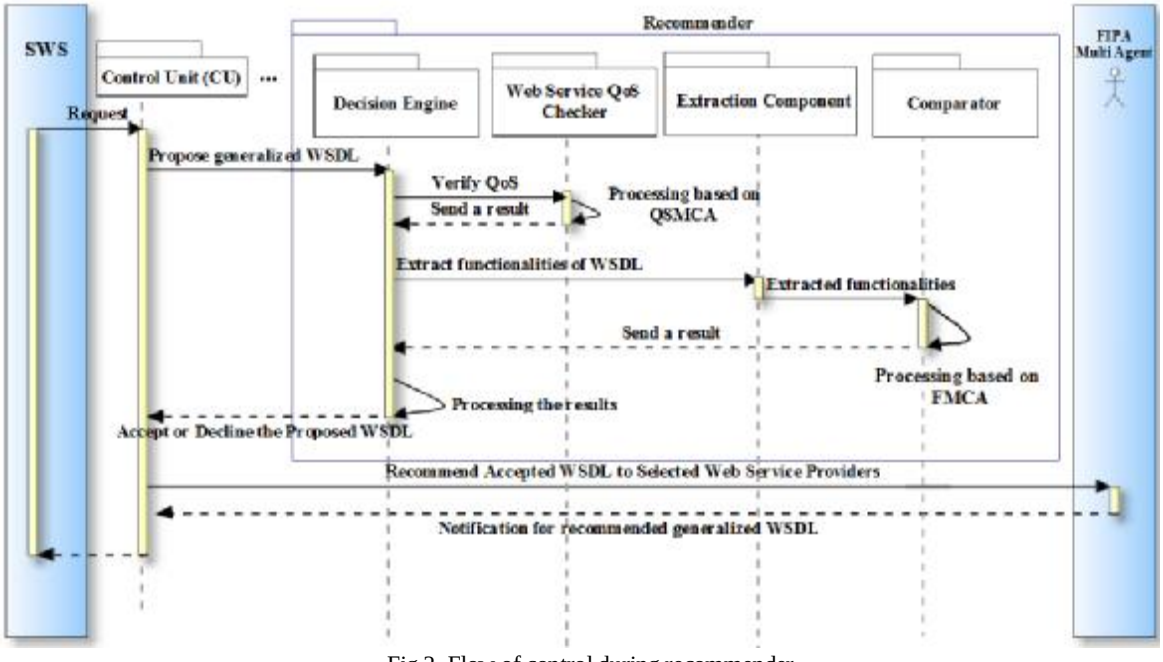

Fig 2. Flow of control during recommender

2.1. Certification Process

In this paper, we introduce a decision algorithm. This algorithm allows decision engine to take part in the process of generalized WSDL certification when recommender wants to evaluate generalized WSDL based on functional and non-functional (QoS) requirements.

According to Fig 3, algorithm FMCA (Functional Metrics Certification Algorithm) handles every functional metric certification of its group dynamically and algorithm QSMCA (QoS Metrics Certification Algorithm) handles every QoS metric certification dynamically based on the current QoS measurement algorithm [12]. At last, it generates the result of acceptance or rejection of the generalized WSDL.

Decision Algorithm

*(1) According to **FMCA**, calculate the match degree of each functionality of generalized WSDL with functionalities of its group in functional registry and return the matched result (FMsg);*
*(2) According to **QSMCA**, calculate the match degree of each QoS property of generalized WSDL with default QoS in QoS registry and return the matched result (QMsg);*
*(3) IF ((FMsg * QMsg)=1)*
(4) Accept=1;
(5) Else
(6) Accept=0;
(7) Return Accept;

Fig 3. Decision algorithm

FMCA (FunctionsList)

1. *Set $_{function}$ ⟵ FunctionsList(N)*
2. *Set $_{registry}$ ⟵ Functional registry $_{function}$(M)*
3. *FC ⟵ 0*
4. *for each item in set$_{function}$*
5. *do if item.num ≥ N*
6. *then*
7. *if item matches item in Set $_{registry}$*
8. *FC ⟵ FC + 1*
9. *else*
10. *add item to Functional registry*
11. *Threshold=FC/M*
12. *If Threshold ≥ 0.5*
13. *FMsg ⟵ 1*
14. *else*
15. *FMsg ⟵ 0*
16. *return FMsg*

Fig 4. FMCA algorithm

As shown in Fig 1, decision engine first sends the generalized WSDL to the extraction component for extracting functionalities of the generalized WSDL. Then the extraction component extracts the FunctionsList. There is a FunctionsList (with N members) to certify that if satisfying the functional requirements as step 1 in Fig 4 and there is a set of functionalities (with M members) of its group in functional registry as step 2. Then as step 5-11, if there are enough records for matching, a counter as FC (Function Count) is set. If no service could match the request, the request will be added into Functional registry. After step 11, the FC will compare with threshold and the FMsg is set and will be sent to the decision engine.

QSMCA (Generalized WSDL)

1. $Set_{QoS} \longleftarrow$ *Generalized WSDL*
2. $QC \longleftarrow$ *QoS Certify*
3. $QC \longleftarrow 0$
4. *for each item in* set_{QoS}
5. *while item.num* $\geq N$
6. *do*
7. *calculate* $QoS_{estimated}$
8. *if* $QoS_{estimated}$ *matches* $QoS_{Registry}$
9. $QC_{item} \longleftarrow 1$
10. *else*
11. $QC_{item} \longleftarrow 0$
12. $QC = QC_1 \wedge QC_2 \wedge \ldots \wedge QC_N$
13. *If QC =1*
14. $QMsg \longleftarrow 1$
15. *else*
16. $QMsg \longleftarrow 0$
17. *return QMsg*

Fig 4. QSMCA algorithm

As shown in Fig 1, decision engine first sends generalized WSDL to the QoS checker for verifying QoS of the generalized WSDL. There is a WSDL to check if satisfying the QoS requirements as step 1 in Fig 5. While there are enough records for estimation, QoS checker will calculate the QoS of this generalized WSDL as step 2- 7 and matches estimated QoS of WSDL with QoS registry. If estimated QoS could match the QoS registry, the QC (QoS Certify) is set as step 8-11. After step 11 the overall of QC will calculate for the generalized WSDL. After verifying QoS, QMsg is set and will be sent to the decision engine.

3. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a new framework for recommender in which evaluates generalized WSDL quality based on functional and non-functional requirements. So that the accuracy of Web service discovery with recommending of generalized WSDL is improved apparently. With this framework, Web Service discovery can be more efficient than before. Evaluation of QoS attributes will be taken into account in the future research. Further, how to deal with composed services is also a challenge in this Web service discovery framework.

4. DETAILS

4.1. Contact adress

Department of computer engineering,
Islamic Azad University of Qazvin,
Qazvin, Iran
E-mail: Shahvalady@qazviniau.ac.ir
Phone: +(98) 9126407360
Fax: +(9821) 22263407

5. ACKNOWLEDGEMENT

I very much appreciated from Dr.mohsenzadeh and Dr.mahmoudi and Mrs Ghari Neiat for helping me for doing my research,Also I am grateful to Dean of Department of computer engineering Islamic Azad University of Qazvin for providing good research's background.

REFRENCES

[1] A. Ghari Neiat , M. Mohsenzadeh, H. Saatchi, S. Haj Shahvaladi: "User Action Ontology-based Semantic Web Service Discovery Framework", proceeding of 2009 International Conference on e-Technology (e-Tech 2009), pp. 3102-3109, 2009 .

[2] A. Ghari Neiat , M. Mohsenzadeh, S. Haj Shahvaladi, A. M. Rahmani," A New Approach for Semantic Web Service Discovery and Propagation based on Agents", Proceedings of the Fifth International Conference on Networking and Services, IEEE, pp. 37-42, 2009 .

[3] A. Mani and A. Nagarajan, "Understanding Quality of Service for Web Services," 2002.

[4] D. Martin, M. Burstein, D. McDermott, S. McIlraith, M. Paolucci, K. Sycara, D. L. McGuiness, E. Sirin, and N. Srinvasan, "Bringing semantics to web services with OWL-S," WORLD WIDE WEB-INTERNET AND WEB INFORMATION SYSTEMS, vol. 10, pp. 243-277, 2007.

[5] I. Foster, N.R. Jennings, C .Kesselman, " Brain meets brawn: Why Grid and agents need each other", Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems. New York, pp. 8-15, 2004.

[6] K. Kritikos, D. Plexousakis , " A Semantic QoS-based Web Service Discovery Algorithm for Over-Constrained Demands ", Third International Conference on Next Generation Web Services Practices, IEEE, pp 49-54, 2007.

[7] L. Ye and B. Zhang, "Web Service Discovery Based on Functional Semantics ", In Proc. of the IEEE Second International Conference on Semantics, Knowledge, and Grid, 2006.

[8] M. Burstein, C. Bussler, M. Zaremba, T. Finin, M. N. Huhns, M. Paolucci, A. P. Sheth, and S. Williams, "A Semantic Web Services Architecture," IEEE INTERNET COMPUTING, vol. 9, pp. 72-81, 2005.

[9] M. Lyell, L. Rosen, L. Casagni-Simkins. Norris, " On software agents and Web services: Usage and design concepts and issues". In Proc. of the 1st International Workshop on Web Services and Agent Based Engineering, Sydney, Australia, 2003.

[10] M. O. Shafiq, Y. Ding, D. Fensel: "Bridging Multi Agent Systems and Web Services: towards interoperability between Software Agents and Semantic Web Services", Proceeding of the 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06), IEEE, 2006.

[11] N. Thio and S. Karunasekera, "Client profiling for QoSbased Web service recommendation," Proceedzngs: 12thAsia-Pacific Soffware Engineering Conference, pp. 8 pp.-8 pp., 2005.

[12] O. Choi., H. Choi , Z. Lee,S. Han." Semantic Web Services Discovery System with QoS for Enhanced Web Services Quality" , CDVE 2007, LNCS 4674, pp. 149–156, 2007.

[13] S. Araban and L. Sterling, "Measuring Quality of Service for Contract Aware Web-Services," in 1st Australian

Workshop on Engineering Service-Oriented Systems (A WESOS 2004). Melbourne, Australia, 2004.

[14] S. Kalepu, S. Krishnaswamy, and S. W. Loke, "Verity: A QoS Metric for Selecting Web Services and Providers," Proceedings of the 4th International Conference on Web Information Systems Engineering Workshops, pp. 13 1-1 39, 2004.

[15] S. Ran, "A Model for Web Services Discovery with QoS," ACMSIGecom Exchanges, vol. 4, pp. 1-10,2003.