



Web service Selection through QoS agent Web service

¹S.Susila, ²S.Vadivel

Department of Computer Science
BITS PILANI DUBAI, UAE

Abstract- The propagation of web services into our businesses and day-to-day lives has made quality of services (QoS) a very important aspect for both the service provider and the consumers. With a large number of web service providers providing similar services, Web service selection based on QoS classification becomes crucial for the consumer. Having discovery agents perform the job of web service selection for the consumer reduces the complexity for the end user.

Keywords: Entropy based Discretization; Quality of Service; Restful web services; extended SOA; Web Services

I. Introduction

With the increasing use of web services in standardization of basic content integration, support of complex service-oriented architectures, provision of seamless integration of business processes and applications etc has lead to a boost in numbers of both web service consumers and providers. This means Quality of Service (QoS) becomes a very important aspect in distinguishing the success of a web service provider. From a consumer's point of view, knowing the QoS provided by the service provider plays a crucial role in choosing a particular web service over its alternatives. Therefore, knowing where the QoS of a web service ranks becomes vital for both the web service provider and the web service consumer.

This has called for tools to measure and classify web services based on various aspects of the service provided. Also, with more and more service providers providing similar functionality the customers job of searching the UDDI registry for a web service that suits his requirements while ensuring desired QoS becomes difficult. In this paper, we aim to design and implement extended SOA which will be able to classify web services based on their QoS and also assist customers in web service discovery based on functionality and QoS. This tool will be then implemented as a web service which will increase ease of use and improved integration. The remainder of the paper is organized as follows: Section 2 presents related work done in selection of web services on the basis of QoS attributes of the web service. Section 3 explains about web service, extended SOA and the QoS of web services in the selection process of web services. Section 4 describes the dataset used for web service selection and the service classification techniques. Section 5 explains Web Service Discovery agent deployed as web service Section 6 reports our results to show that our feature selection method is effective in the selection of web service and finally we conclude in chapter 7.

II. Related work

Selection of Web Services on the Basis of Quality of Service Constraints was proposed in the paper [1]. In that QoS Manager had a role of being a moderator amongst the provider and the client. In paper [2] they discuss about QoS broker publish system that Extract the quality of service constraints in the issued WSDL and the values extracted are stored in QoS DB and the fundamental features are issued in the UDDI registry and service matching procedure is applied, and finally, service with the highest quality selected and proposed to the service requester. In paper[8] they formulates a robust QoS semantic framework for Web Services into three layers QoS-ontology, which can provide a standard model to formally describe arbitrary QoS parameters and exhibits properties [3][4][5]. Paper [6] dealt with, prominent works that apply fuzzy theory for representing imprecise QoS constraints and preferences and for developing QoS based ranking algorithm for Web services which can deal with fuzzy QoS values. In [7], the service selection for a Web service composition problem PWSDCP is dignified as contentment difficulty that belongs to a fuzzy attribute. Every QoS criterion has 5 fuzzy sets which describes the intensity of the attribute.

III. Extended SOA

Web Services are outcome of the advancement of the web into a means of scientific, commercial and social exchanges. A Web service can be described as a way of calling a function which is inside software from software. The software which makes the call is called as the client and the software which services the client is called a server. The two softwares might have been programmed using different languages and could be running on different machines but have to be connected by a network. Web services have an interface expressed as the WSDL (Web services description language) file. WSDL file can be seen a contract for communications between the web service client and server.

The Service Oriented Architecture (SOA) consists of three basic blocks namely service registry, service consumer and service provider. The Extended SOA has another block known as Agent in this case which helps

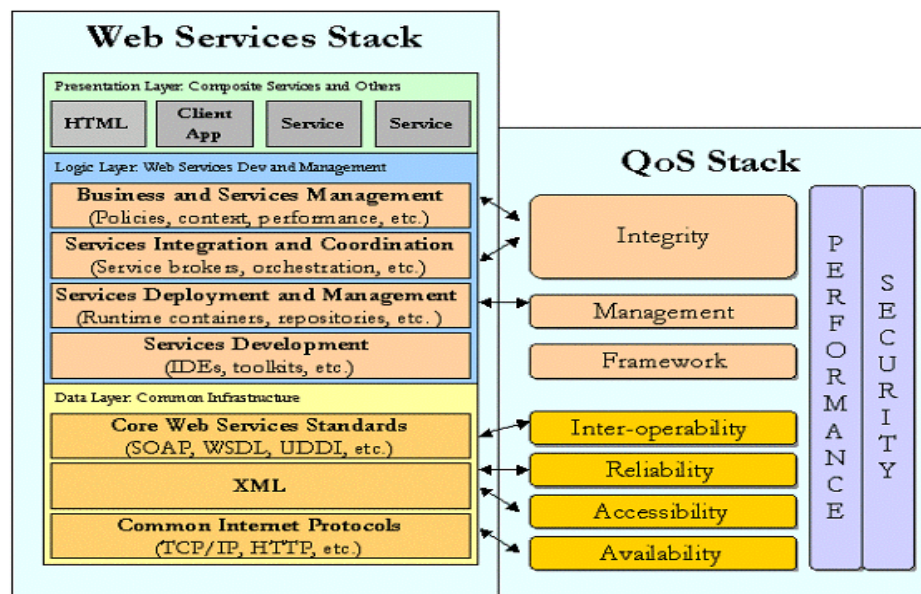
in the selection of web services based on their qos attributes. The discovery agent web service is intended to make web service consumer's job of finding a web service suitable to his business requirements and performance needs easier. This web service takes in the web services needed and the QoS requirements of the user and gives out recommendations to the user in the form of WSDL links.

With the rapid growth in the field of web services, Quality of service (QoS) becomes one of the important factors based on which the service requestor distinguishes the success of service providers [5]. Some of the aspects of QoS include response time, reliability, scalability, throughput, robustness, success ability, exception handling, reliability, accuracy, integrity, accessibility, availability, interoperability, security, and network-related QoS requirements.

Figure 1 An Extended SOA for web service selection.



Figure 2. QoS Stack along with layer of web services stack. [10]



The above figure shows the relationship between various elements of the web service stack to the QoS attributes in the QoS stack. The aim of us is to classify the web services based on the response time, availability, throughput, success ability and reliability and hence help the service requestor choose a web service which best suits his requirements.

IV. Data set and service classification

The input data set used for the decision tree induction and QoS classification is the "QWS Dataset" [8][9][10]. The QWS dataset consists of data from over 5000 web services out of which the public dataset consists of a random 365 web services which have been chosen and nine QWS(Quality of Web Service) attributes have been measured. Each web service was tested for over duration of over ten-minutes for three successive days.

Figure 3. QWS Parameters with their description and units.

ID	Parameter Name	Description	Units
1	Response Time	Time taken to send a request and receive a response	ms
2	Availability	Number of successful invocations/total invocations	%
3	Throughput	Total Number of invocations for a given period of time	invokes/second
4	Successability	Number of response / number of request messages	%
5	Reliability	Ratio of the number of error messages to total messages	%
6	Compliance	The extent to which a WSDL document follows WSDL specification	%
7	Best Practices	The extent to which a Web service follows WS-I Basic Profile	%
8	Latency	Time taken for the server to process a given request	ms
9	Documentation	Measure of documentation (i.e. description tags) in WSDL	%
10	WsRF	Web Service Relevancy Function: a rank for Web Service Quality	%
11	Service Classification	Levels representing service offering qualities (1 through 4)	Classifier
12	Service Name	Name of the Web service	None
13	WSDL Address	Location of the Web Service Definition Language (WSDL) file on the Web	None

Mostly, all of the quality of service constraints varies from one another in direction as well as in value range of the utility increments. There is no comparison between them. Therefore, calculation of the weighted average of quality of service constraints is not useful. Constraint values must be transformed such that they reflect the true value in a standard range and also providing the same incrementing direction. Let's say that raw value of constraint, Q , is denoted by q , threshold value is denoted by q_{th} and q_{min} denotes the minimum [3].

Data normalization of a constraint is calculated according to equation (1) if the effectiveness of it increases with the value of the constraint, q . Or else, equation (2) is applied.

$$Q' = \frac{(q - q_{min})}{(q_{max} - q_{min})}, \quad \text{if } q_{max} - q_{min} \neq 0 \quad (1)$$

$$Q = \frac{(q_{th} - q)}{(q_{th} - q_{min})}, \quad \text{if } q_{th} - q_{min} \neq 0 \quad (2)$$

On the whole, final rank value WSRF (web service relevancy function) is calculated using weighted sum of the quality of service constraints which were normalized, according to equation (3) and we get dataset.

$$V = \sum_{i=1}^n W_i \times Q_i \quad (3)$$

The service classification characterizes different levels of service contributing qualities. There are four service classifications:

1. Excellent (High quality)
2. Good
3. Average
4. Poor (Low quality)

The classification is differentiated on the on the whole quality evaluation calculated by WsRF. Using WsRF values found for every Web service, we apply a classification format to relate each Web services to a particular service group. The classification can be useful to distinguish between ranges of services that offer the similar functionality. The part of the dataset is shown in figure.

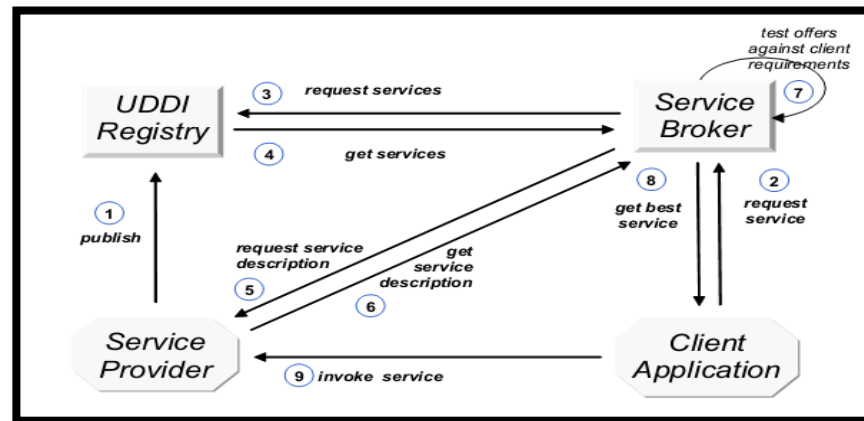
Figure 4 Dataset

S. No.	Response Time (ms)	Availability (%)	Throughput (invokes/second)	Successability (%)	Reliability (%)	Compliance (%)	Best Practices (%)	Latency (ms)	Documentation (%)	WsRF	Classification
1	3321.4	89	1.4	96	73	78	80	2.6	96	-0.00053	POOR
2	302.75	89	7.1	90	73	78	80	187.75	32	0.00051	AVERAGE
3	482	85	1.6	95	73	100	84	1	2	0.000713	AVERAGE
4	126.17	98	12	100	67	78	82	22.77	89	0.001279	GOOD
5	107	87	1.9	95	73	89	62	58.33	93	0.000911	AVERAGE
6	107.57	80	1.7	81	67	78	82	18.21	61	0.000803	AVERAGE
7	255	98	1.3	99	67	100	82	40.8	4	0.000439	AVERAGE
8	136.71	76	2.8	76	60	89	69	11.57	8	0.000503	AVERAGE
9	102.62	91	15.3	97	67	78	82	0.93	91	0.001428	GOOD
10	93.37	96	13.5	99	67	89	58	41.66	93	0.00129	GOOD
11	133	86	7.7	95	73	78	84	10.67	9	0.000676	AVERAGE
12	221.48	90	10.9	97	53	89	66	37.26	6	0.000681	AVERAGE

V. Web Service Discovery agent deployed as web service.

The discovery agent web service is intended to make consumer's job of finding a web service suitable to his business requirements and performance needs. This web service takes in the web services needed and the QoS requirements of the user and gives out recommendations to the user in the form of WSDL links. This design works as per the broker based architecture where in we have a web service broker/agent which checks for the consumer's requirements and looks up for a service provider who matches the said requirements. The below figure depicts the flow of requests and responses between the entities involved.

Figure 5. Interaction between the participating entities in agent/ broker based architecture.



This web service access the Java Derby database where a database consisting of QoS Parameters along with the web service description and WSDL are stored using the Java Persistence API (JPA).

Figure 6. Snapshot of the Java Derby database used for testing.

jdbc:derby://localhost:1527/QoSDB [app on APP]

APP

Tables

QOS

QOSDB

Views

Procedures

NULLID

SQLJ

SYS

SYSCAT

SYSCS_DIAG

SYSCS_UTIL

SYSFUN

SYSIBM

SYSPROC

SYSSTAT

jdbc:derby://localhost:1527/sample [app on APP]

Web Services

Servers

Hudson Builders

select * from APP.QOSDB

Page Size: 20

Total Rows: 10 Page: 1 of 1 Matching Rows:

#	AVAILABILITY	RESPONSETIME	THROUGHPUT	SUCCESSABILITY	RELIABILITY	CLASSIFICATION	SERVICE	WSDL
1	332	89	1	96	73 poor	map	http://xmla.nig.ac.jp/vsd/Mapping.wsdl	
2	302	89	7	90	73 average	time	http://www.vbrain.net/samplewebprojects...	
3	482	85	16	95	73 average	mail	http://ws.xwebservices.com/XWebEmailVal...	
4	126	98	12	100	67 good	map	http://xml.assessment.com/service/MAPPR...	
5	107	87	1	95	73 average	mail	http://vm2mail.vmttransfer.com/vs/SendW...	
6	107	80	1	81	67 average	map	http://maps.ers.usda.gov/Profiles/ERSWeb...	
7	102	91	15	97	67 good	time	http://markitup.com/WebServices/TimeZon...	
8	93	96	13	99	67 good	mail	http://ws.xwebservices.com/XWebEmailVal...	
9	337	70	15	100	80 excellent	map	http://www.viamichelin.com/vs/services/G...	
10	335	66	16	100	82 excellent	time	http://dotnet.jku.at/buch/samples/7/encodi...	

Usages

Output

Test Results

Java DB Database Process

SQL Command execution

Mon Dec 24 21:18:32 GST 2012 : Security manager installed using the Basic server security policy.

Mon Dec 24 21:18:35 GST 2012 : Apache Derby Network Server - 10.8.1.2 - (1095077) started and ready to accept connections

The web service is deployed on glassfish server V3. The web service operates using SOAP protocol. On successful deployment, the web service is then tested using the IDE's tester page which runs on the local browser.

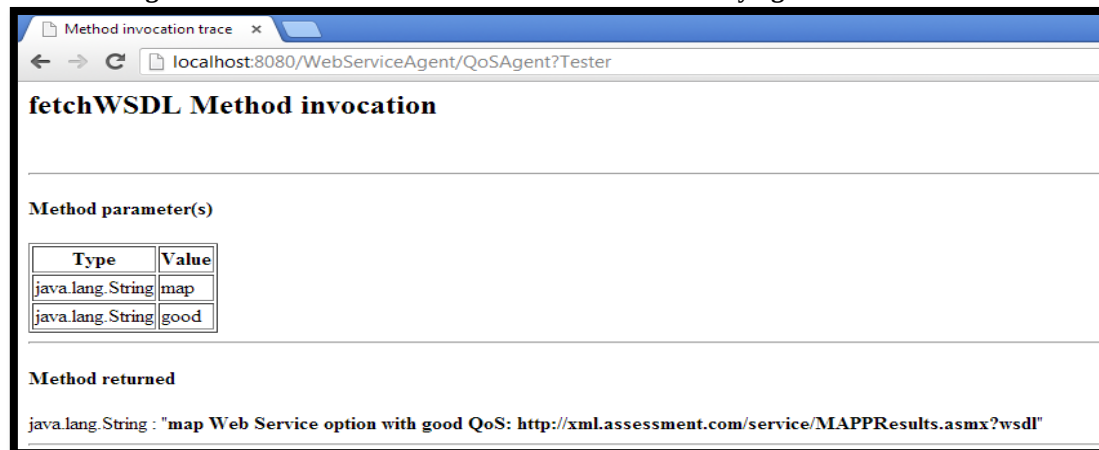
VI. Discovery agent web service

In this we have considered three web services specific to their functional domains namely map, time and mail web services. As shown in the fig., there is a database maintained by the Agent which helps the client to choose a web service of his requirement. The client is asked to enter the functional domain and the kind of web service on the basis of non functional attributes, as explained before there are four types of web services available in our prototype namely Excellent (High quality), Good, Average and Poor (Low quality). At times client may want Average or even poor web service in terms of cost attribute, where he is compromised with other attributes like response time, availability, throughput etc. Then the agent web service returns wsdl links of those web services which are tailored for that particular client requirement as he click on fetch WSDL button.

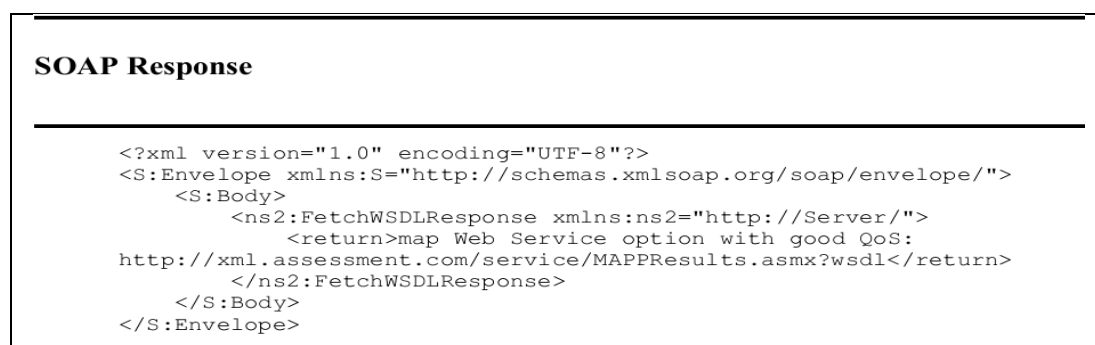
Figure 7. IDE's tester page for the discovery agent web service.



Figure 8 IDE method invocation trace for the discovery agent web service



Clippings of the SOAP request, response and the WSDL file the discovery agent web service is given below,



VII. Conclusion

Web Services are becoming increasingly used and a large number of consumers are building their business solutions using web service technology. The need for Qos specifications for web services has arisen due to consumer's prospect for superior web service performance. Services provider's obligation to provide high quality service so as to improve the usability and utility of their services which in turn decides their standing in the market. A service discovery agent was implemented and deployed as a web service. Even though this implementation isn't a full-fledged one, it does give us an intuitive overview of a service discovery agent to enable us understand the basic working of such a system.

References

- [1] Manish Godse, UmeshBellur, Rajendra Sonar, "Automating QoS based Service Selection", 2010 IEEE International Conference on Web Services PP: 530-540.
- [2] KyriakosKritikos and DimitrisPlexousakis, "Requirements for QoS Based Web Service Description and Discovery", IEEE Transactions on Services Computing, Vol. 2, 2009.
- [3] Qu Li-li, Chen Yan, "Qos Ontology Based Efficient Web Services Selection", International Conference on Management Science & Engineering, 2009.
- [4] Serhani M. A., Dssouli R., Hafid A., Sahraoui H., "A QoS broker based architecture for efficient web services selection", IEEE International Conference on Web Services, 2005.
- [5] Yu T., Lin K. J., "Service selection algorithms for Web services with end-to-end Qos constraints", Proceeding of Information Systems and E-Business Management, 2005.
- [6] Tran VuongXuan, Tsuji Hidekazu, "QoS based ranking for Web Services: Fuzzy Approach", International Conference on Next Generation Web Services Practices, 2008.
- [7] Lin M., Xie J., Guo H., Wang H., "Solving Qos-Driven Web Service Dynamic Composition as Fuzzy Constraint Satisfaction", IEEE International Conference on E-Technology, E-Commerce and E-Service, 2005.
- [8] Al-Masri, E., and Mahmoud, Q. H., "Discovering the best web service", (poster) 16th International Conference on World Wide Web (WWW), 2007, pp. 1257-1258. (for QWS Dataset Version 1.0 or QWS Dataset Version 2.0)
- [9] Al-Masri, E., and Mahmoud, Q. H., "QoS-based Discovery and Ranking of Web Services", IEEE 16th International Conference on Computer Communications and or QWS Dataset Version 2.0)
- [10] Al-Masri, E., and Mahmoud, Q.H., "Investigating Web Services on the World Wide Web", 17th International Conference on World Wide Web (WWW), Beijing, April 2008, pp. 795-804. (for QWS-WSDLs Dataset Version 1.0)