

# Discovering the Best Web Service

Eyhab Al-Masri and Qusay H. Mahmoud  
 Department of Computing and Information Science  
 University of Guelph, Guelph, ON, Canada N1G 2W1  
 {ealmasri, qmahmoud}@uoguelph.ca

## ABSTRACT

Major research challenges in discovering Web services include, provisioning of services across multiple or heterogeneous registries, differentiating between services that share similar functionalities, improving end-to-end Quality of Service (QoS), and enabling clients to customize the discovery process. Proliferation and interoperability of this multitude of Web services have led to the emergence of new standards on how services can be published, discovered, or used (i.e. UDDI, WSDL, SOAP). Such standards can potentially provide many of these features and much more, however, there are technical challenges associated with existing standards. One of these challenges is the client's ability to control the discovery process across accessible service registries for finding services of interest. This work proposes a solution to this problem and introduces the Web Service Relevancy Function (WsRF) used for measuring the relevancy ranking of a particular Web service based on QoS metrics and client preferences. We present experimental validation, results, and analysis of the presented ideas.

## Categories and Subject Descriptors

D.2.12 [Software Engineering]: Interoperability – *data mapping, distributed objects, interface definition languages*; H.3.5 [Information Storage and Retrieval]: Online Information Services – *data sharing, Web-based services*

**General Terms:** Design, Management, Measurement, Performance, Theory, Verification

**Keywords:** UDDI, UDDI Business Registries, Quality of Services, QoS, Web Services, UDDI Extension, tModel

## 1. INTRODUCTION

In recent years, Web service technologies have demonstrated usefulness and promising capabilities in deploying understandable applications used for business-to-business integration. However, Web services' adoption rate is still slow [1] due to many challenges. Some of these challenges include: (1) inexistence of reliable approaches for computing and monitoring QoS properties in a fair and transparent manner to both service requesters and providers, and (2) inability to associate QoS parameters into the discovery process.

In addition, standards such as UDDI have enabled service providers and requesters to publish and find services of interest through UDDI Business Registries (UBRs), respectively. However, UBRs are not adequate enough for enabling clients to effectively find relevant Web services due to a variety of reasons.

As Web services proliferate, size and magnitude of UBRs are likely to increase significantly, and hence searching multiple UBRs will become very time consuming and inefficient using simple keyword-based search operations.

The UDDI specification V3 [2] does not include QoS as part of its publication or inquiry APIs. In addition, it does not provide the ability to associate QoS information within the registry through tModels, or bindingTemplates in a standardized manner. Unfortunately, this major design deficiency hinders service requesters from being able to define any search criteria based on QoS parameters or differentiate and compare between services prior to any invocations or bindings during the discovery process.

To address the above issues, this work introduces a mechanism that extends our Web Service Repository Builder (WSRB) architecture [3] by offering QoS support for Web services and uses a combination of service parameters as constraints when searching for relevant Web services. Our solution has been tested and results show high success rate in providing the correct or most relevant Web service of interest within top results.

## 2. WEB SERVICE QOS MANAGER (WS-QoSMan)

The main focus of our approach is to design an intelligent system that has the potential of examining Web service's QoS properties in an open and transparent manner, and enabling clients to select the best available Web service by taking advantage of client QoS preferences, Web service capabilities, and service provider features. This is achieved through the WS-QoSMan service broker. The architecture of the proposed WS-QoSMan solution is shown on Figure 1.

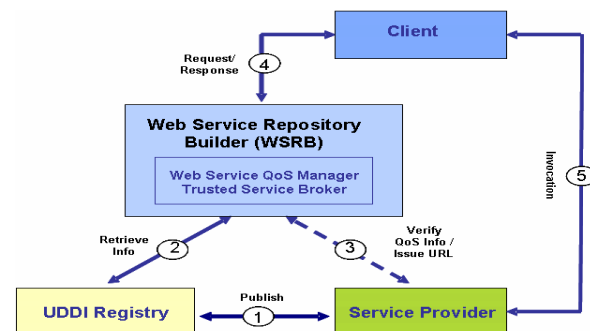


Figure 1. Web Service Quality of Service Manager (WS-QoSMan) Architecture.

The WS-QoSMan applies an external resource approach in which it uses a tModel called QoSMetrics that contains information to an external reference. This is very similar to the tModel used for pointing to WSDL files. QoSMetrics uses overviewURL to point to an XML-based file generated by WS-QoSMan and that contains QoS metrics for a specific Web service.

### 3. WS-QoSMan COMPUTATION

In order to measure the relevancy of Web services to a given query, WS-QoSMan uses a Web Service Relevancy Function (WsRF). WsRF is used to measure the relevancy ranking of a particular Web service  $ws_i$ . Clients can submit their requests to WS-QoSMan (i.e. via a GUI) which will process these requests and compute WsRF values for all available services related to search query. A Web service with the highest calculated WsRF value is the most desirable and relevant to the client based on his/her preferences. In order to calculate  $WsRF(ws_i)$ , we need the maximum normalized value for each set of QoS parameters. Let  $N$  be an array where  $N = \{n_1, n_2, n_3, \dots, n_m\}$  with  $1 \leq m \leq I$  such that:

$$N(j) = \sum_m^j q_{m,j} \quad (1)$$

where  $q_{m,j}$  represents QoS measured value of a particular  $ws_i$  and QoS parameter. Each QoS measured value is compared against the maximum in its corresponding set based Equation 2 such that:

$$h_{i,j} = w_j \left[ \frac{q_{i,j}}{\max(N(j))} \right] \quad (2)$$

where  $h_{i,j}$  measures the distance of  $q_{i,j}$  from the maximum normalized value in the corresponding QoS property group or  $j$  column, and  $w_j$  represents the weight associated with a QoS parameter (provided by the client). Once the comparison takes place, WsRF is calculated as shown on Equation 3:

$$WsRF(ws_i) = \sum_{i=1}^N h_{i,j} \quad (3)$$

where  $N$  represents the number of Web services from a given set.

### 4. EXPERIMENTS AND RESULTS

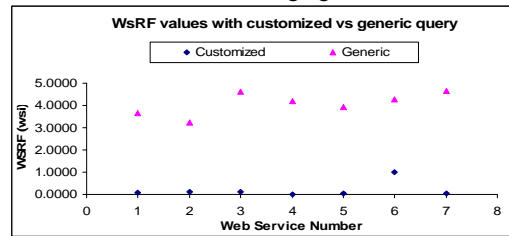
Data used in this work are based on seven actual implementations for email verification Web services that exist over the Web and are listed in XMethods.net, XMLLogic, and StrikeIron. Definitions and measurements of QoS parameters were based on other studies reported in literature [4,5] including: Response Time (RT), Throughput (TP), Availability (AV), Accessibility (AC), Interoperability Analysis (IA), and Cost (C). Results of WS-QoSMan measurements are shown on Table 1. WS-QoSMan used MindReef's SOAPScope Analysis feature for measuring IA while cost was provided by service providers.

**Table 1. QoS Metrics for available email verification services**

ID	Service Provider	RT (ms)	TP (req./min)	AV (%)	AC (%)	IA (%)	C (cents/invoke)
1	XMLLogic	720	6.00	85	87	80	1.2
2	XWebservices	1100	1.74	81	79	100	1
3	StrikeIron	710	12.00	98	96	100	1
4	StrikeIron	912	10.00	96	94	100	7
5	CDYNE	910	11.00	90	91	70	2
6	Webservicex	1232	4.00	87	83	90	0
7	ServiceObjects	391	9.00	99	99	90	5

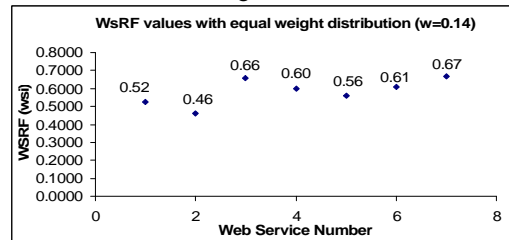
Figure 2 shows results from running WsRF on Web services listed in Table 1 when performing a customized search query that is heavily dependent on cost vs. a generic search query. The WsRF

values when associated with at least one weight of any QoS parameter results in values ranging from 0 to 1 while having a broader search that is not QoS specific (i.e. without associating weights) produces values ranging from 3.22 to 4.67. Having smaller WsRF values means that standard deviation is smaller and therefore, WsRF is closer to converging into a solution.



**Figure 2. Results from running WsRF heavily dependent on cost vs. generic search (w/o weights).**

To effectively test for accuracy, another test was conducted by equally distributing weights across all QoS parameters ( $w=0.143$ ) and yields results shown on Figure 3.



**Figure 3. WsRF values when distributing weights across QoS parameters equally ( $w=0.143$ ).**

### 5. CONCLUSION

A Web service relevancy ranking function (WsRF) based on QoS parameters has been presented in this paper for the purpose of finding the best available Web service during Web services' discovery process based on a set of given client QoS preferences or QoS search criteria. QoS parameters from accessible Web services were measured using WS-QoSMan and were used as search constraints in order to retrieve Web services with an accurate relevancy ranking. The proposed solution provides an effective Web service relevancy function that is used for ranking and finding most relevant Web services. For future work, we plan to extend QoS parameters to include information such as reputation, penalty rates, compliance, reliability, and fault rates.

### 6. REFERENCES

- [1] UDDI Spec Technical Committee, "UDDI Version 3.0.2", [http://uddi.org/pubs/uddi\\_v3.htm](http://uddi.org/pubs/uddi_v3.htm), 2003.
- [2] Adam, C., From Web Services to SOA and Everything in Between: The Journey Begins, Webservices.org, May 2005.
- [3] Al-Masri, E., and Mahmoud, Q.H., A Framework for Efficient Discovery of Web Services across Heterogeneous Registries, IEEE Consumer Communication and Networking Conference (CCNC) 2007.
- [4] Ran, S., A Model for Web Services Discovery With QoS, ACM SIGecom Exchanges 4(1):1-10, 2003.
- [5] Kalepu, S., Krishnaswamy, S., Loke, S. W., Verity: A QoS Metric for Selecting Web Services and Providers", WISEW pp.131-139, 2003.