An Efficient Framework for Agent-Based Quality Driven Web Services Discovery

T. Rajendran

Assistant Professor cum Research Scholar,
Department of Computer Science &Engineering
SNS College of Technology, Coimbatore, India,
E-Mail: rajendran_tm@yahoo.com

P. Balasubramanie

Professor, Department of Computer Science & Engineering
Kongu Engineering College, Perundurai, India,
E-Mail**:** pbalu_20032001@yahoo.co.in

***Abstract*— Web service technology is playing a major role in today's distributed system computing. The wide adoption of web services raises the challenging problem of service discovery. With an increasing number of Web services providing similar functionalities, Quality of Service (QoS) is becoming an important criterion for selecting of the best available service. We aim to refine the discovery process through designing a new framework that enhances retrieval algorithms by combining syntactic and semantic matching of services. We propose a model of QoS-based Web services discovery that combines an augmented UDDI registry to publish the QoS information.**

Keywords- Web Services discovery; Quality of Services (QoS); Web Service Agent; Service Registry; UDDI; WSDL; SOAP; tModel

I. INTRODUCTION

Services are means for building distributed applications and are used to build service-based applications. Service-oriented computing is a general topic, more specifically in the context of WWW technologies we are speaking about Web Services. In last few years we can observe the increasing popularity of web services. SOA (Service Oriented Architecture) is an approach to build distributed systems that deliver application functionality as services which are language and platform independent. A web service is a technology that realizes the SOA. The current Web services architecture encompasses three roles: Web service provider, Web service consumer and Universal Description, Discovery and Integration (UDDI). The Web service provider publishes a description of the service in the UDDI registry. Users (service consumers) will search that directory to get their desired services. UDDI registry includes businessEntity, businessService bindingTemplate, and tModel data structure [1]. In UDDI, information about businesses and services is described in XML. Search queries on UDDI are processed according to keywords and categorization information. Although UDDI has been widely accepted as a web infrastructure standard, there is limitation with its discovery mechanism. There are two major problems in using QoS for service discovery. First is the specification and storage of the QoS information, and second is the specification of the customer's requirements and matching these against the information available.

The discovery process falls in two steps: Web Service matching which is meeting specific functionality required by a consumer (user) from existing services. Web Service selection which is choosing a service with the best quality among those matched services. Now, with an increasing number of Web services providing similar functionalities, more emphasis is being placed on how to find the service that best fits the web service consumer's requirements which are functional requirements (what the service can do) , and nonfunctional requirements, such as the price and quality of service guaranteed by a service provider[2,3]. We propose a Web services discovery model that contains an extended UDDI to accommodate the QoS information.

The rest of the paper is structured as follows. Our proposed discovery framework is illustrated in Section II. Finally the conclusion and the future work are discussed in Section III.

II.THE PROPOSED AGENT-BASED FRAMEWORK FOR QUALITY WEB SERVICES DISCOVERY

We propose a framework for agent-based web service discovery with QoS for an objective of selecting the best web service that satisfies consumers QoS constraints and preferences. The framework involves four main participating roles. They are Web Service Consumer (client), Web Service Provider, Web Service Agent (WSA), in addition to a QoS enabled augmented UDDI registry using tModel data structure. Components of the framework are presented in figure 1. The Web Service Provider is the entity that develops the web service and describes its functionalities in addition to the QoS it provides. Web service consumer normally expects requirements on functionally similar and QoS are to be satisfied by the Web services.

A Web service Agent does the following functions:

- To receive messages containing the requested service functionality along with QoS constraints from the requester.
- To discover functionally similar Web services from the Web service registry through functionality matching [4,5].
- To select the most suitable Web service based on the requester's QoS constraints and preferences.

978-1-4244-4711-4/09/$25.00 ©2009 IEEE

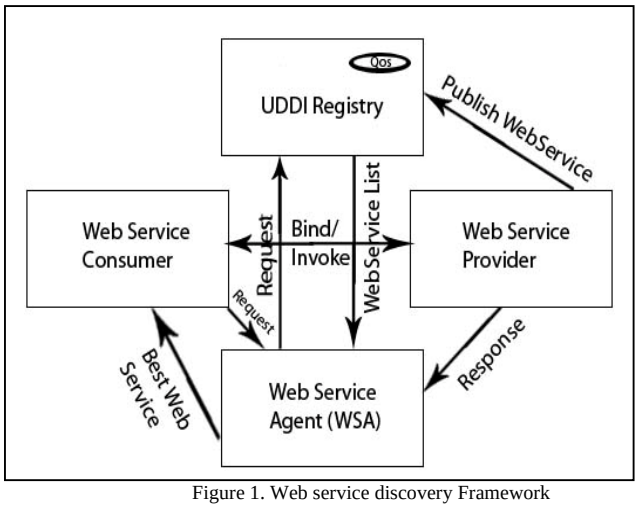

Figure 1. Web service discovery Framework

The WSA assists clients in selecting web services based on a set of QoS parameters. The agent is a web services performing a collection of QoS functionalities. It is the entity that performs the verification and certification tasks. It is also involved in other operations, such as QoS negotiation, monitoring, and adaptation. The WSA holds up-to-date information on offers currently available for a group of services which have been requested recently. Offers are grouped by the interface (tModel) that the services providing them implement. The first time a Web service for a certain interface is requested, one or more UDDI registries associated with the Web Service Agent are requested. The WSDL files for these services are then checked for QoS extensions and available offers are built. From then on this newly created offer list is consulted in order to find the best match for clients and their requirements, allowing an accelerated lookup process. To keep offer lists up-to-date, the WSA queries the UDDI periodically in order to check for new offers. Once an offer expires, it is deleted from the WSA's local cache. If the validity of the offer is extended, it will be re-detected during the next check.

When a client application inquires the WSA for the cheapest available offer, it sends its QoS requirements as a part of the request. In the order of their price, the WSA then tests the available offers whether they fulfill the consumer's requirements. The first compliant offer is returned to the client. It is worth noting that one can implement her own strategy for defining the QoS parameters and the selection of the appropriate services. We just give here a simple idea of how the selection could be done.

The implementation of WSA uses a remote Web Service to obtain the access point of the most appropriate service. This version is mainly intended as a light process for multiple client applications that use a single private WSA that runs as a Web service within their network domain. Before service invocation, the client application will inquire the WSA for the most appropriate offer available that fulfills the requirements. The WSA selects the most appropriate offer on behalf of the client from the WSA's local database. (Note we assume in this case that the WSA has already a local cache of the services the client is asking for. This model ensures a fast response time.) Once the client gets the required offer from the WSA, the client will invoke the service with the desired QoS properties. The QoS properties are transmitted in the SOAP headers to the service provider that can treat the request based on the QoS properties. Therefore, service offers are normally analyzed in advance and testing against consumer requirements can be performed immediately on receiving a request.

A. QoS parameters for web services

QoS for web services represents the non-functional aspects of the service being provided to the web service users. For the sake of our experiments, we will consider the following QoS:

- *Price:* is the cost involved in requesting the service. The web service cost can be estimated by operation or by volume of data.
- *Response time*: is the time a service takes to respond to the client request.
- *Availability:* the probability that the service is accessible or the percentage of time that the service is operating.
- *Throughput:* is the rate at which a service can process requests attributes.

III. CONCLUSION

In this paper we have presented a new approach for Web Service discovery process. Due to the increasing popularity of Web services technology and the potential of dynamic service discovery and integration, multiple service providers are now providing similar services. QoS is a decisive factor to distinguish functionally similar Web services. Our suggested theoretical framework will be based and implemented on QoS properties. To achieve our goal we need to architect our new model and start developing an agent-based framework to support QoS aware discovery of web services, develop the matching algorithm that exploits syntactic and semantic information from WSDL specifications. An amount of services is needed where we can test the performance of our system. We need to simulate the matching process under different IR models to select the best model. This will enable a more flexible, and trustable architecture. Results of this work will be reported in a future paper.

REFERENCE

[1] UDDI Technical White Paper", from http://uddi.org/pubs/uddi tech-wp.pdf, 2000.

[2] D. Gouscos, M. Kalikakis, and P. Georgiadis, "An Approach to Modeling Web Service QoS and Provision Price", Proceedings of the 1st International Web Services Quality Workshop WQW 2003 at WISE 2003, Rome,Italy, pp.1-10, 2003.

[3] S. Ran, "A Model for Web Services Discovery with QoS", ACM SIGEcom Exchanges, Vol. 4(1), pp.1–10, 2003.

[4] H. Kreger, " Web Services Conceptual Architecture (WSCA1.0)", Published in May 2001, http://www.ibm.com/software/solutions/ webservices/ pdf/wsca.pdf.

[5] Y. Wang and E. Stroulia, "Flexible Interface Matching for Web-Service Discovery", Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE'03), IEEE 2003.