Journal of Theoretical and Applied Information Technology
15 June 2012. Vol. 40 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN INTEGRATED APPROACH TO RAPID AUTOMATED SERVICE DISCOVERY OF SEMANTIC WEB SERVICE

[1] MICHAEL RAJ T.F
[1]Asstt Prof., Department of Computer Science Engineering, SASTRA University
E-mail: [1]tfmichaelraj23@yahoo.com

ABSTRACT

The development of Web services has transformed the World Wide Web into a more application-aware information portal. Web services are interoperable and extensible and there are possibilities for simple services to be clustered to build complex ones. To improve the automation of web services selection a lot of techniques and technologies are recommended. Web service selection in a web service group is an emerging research topic. In this paper we propose an integrated model for web service selection and clustering can be done from a web service group. The proposed model builds up a lightweight integrated model.

Keywords: *Semantic Web Services(SWS),Clustering Web Services, Ontologies*

1. INTRODUCTION

With the rapid increase of web services in the internet it is becoming more and more difficult to locate the proper web services in the greatest number of services. To locate the proper services efficiently become very important. One of the web services can be easily selected among the no of services is very simple. As the set of available web services expands it is becomes increasingly important to have efficient approach to help identify services that match a requester's requirements [1].

Processing the pool of service group, how to quickly and accurately find the services to meet the need which influences the further development of the web services. Web services may be used for business applications or in government and military. However this requires careful selection and composition of appropriate web services. The web services within the service registry UDDI[2] have predefined categories that are specified by the service providers. As a result similar services may be listed under different categories. Given the large number of web services and the distribution of similar services in multiple categories in the existing UDDI infrastructure, it is difficult to find services that satisfy the desired functionality. So there is a need to categories web services based on their functional semantics rather than based on the classifications of the service providers.

In order to address the limitations of existing approaches, an integrated approach needs to be developed for addressing the two major issues related to automated service discovery: 1) semantic based categorization of Web Services; and 2) selection of services based on semantic service description rather than syntactic keyword matching. Moreover, the approach needs to be generic and should not be tied to a specific description language. Thus, any given Web Service could be described using WSDL, OWL-S[3], or through other means. Furthermore, the approach should make no assumptions about the kinds of Web Services. In specific, we do not make any assumption about whether the Web Services are developed in house or offered to users by third party service providers. In this model we adopt an ontology based service discovery method which is based on the calculation of similarity of services, inputs, outputs and functional ontology and clustering the WS.

The rest of the paper is organized as follows. Section 2 introduces related concepts to be used throughout the remainder of this paper. Section 3 provides related work to introduce an integrated model including the simple services matchmaking. Finally, Section 4 concludes the paper.

2. RELATED CONCEPTS

2.1 Web service

Web service is interoperable unit of application logic that transcends programming

Journal of Theoretical and Applied Information Technology
15 June 2012. Vol. 40 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

languages, Operating systems, network communication protocols such as TCP/IP [4] and data representation issues

The World Wide Web has evolved from a static information repository to a current dynamic distributed information sharing and processing source. Web Services [5, 6] are one of the latest endeavors in this evolution. Together with layers of XML-based open standards [7], Web Services provide a framework for automated service advertisement, discovery, invocation, composition &inter-operation and execution monitoring.

Web services are based on the following industry standards extensible mark-up language(XML),simple object access protocol(SOAP),Web service description language(WSDL) and universal description discovery and integration(UDDI).

Web service can be expressed as four tuples as follows

Webservice={Name(N),Description(D),Inputs(I),Outputs(O)}

I –set of inputs $\{i_0, i_1, i_2, i_{3,\ldots\ldots}\ i_m\}$

O –set of outputs $\{o_0, o_1, o_2, o_{3,\ldots\ldots}\ i_n\}$

2.2 Web Service Architecture

There are two ways to view the web service architecture (Figure 1). The first is to examine the individual roles of each web service actor; the second is to examine the emerging web service protocol stack.

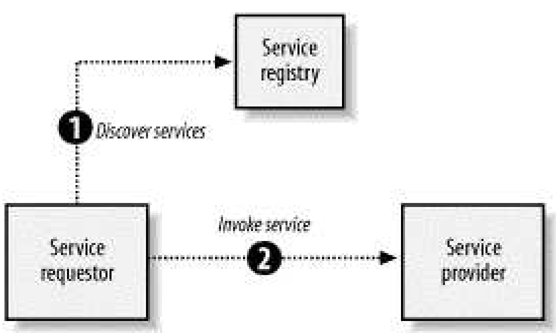


Fig. 1. Web service architecture

2.2.1 Web Service Roles

There are three major roles within the web service architecture:

2.2.2 Service provider

This is the provider of the web service. The service provider implements the service and makes it available on the Internet.

2.2.3 Service requestor

This is any consumer of the web service. The requestor utilizes an existing web service by opening a network connection and sending an XML request.

2.2.4 Service registry

This is a logically centralized directory of services. The registry provides a central place where developers can publish new services or find existing ones. It therefore serves as a centralized clearinghouse for companies and their services.

2.3 Web Service Protocol Stack

A second option for viewing the web service architecture is to examine the emerging web Service protocol stack. The stack is still evolving, but currently has four main layers. Following is a brief description of each layer.

2.3.1 Service transport

This layer is responsible for transporting messages between applications. Currently, this layer includes hypertext transfer protocol (HTTP), Simple Mail Transfer Protocol (SMTP), file transfer protocol (FTP), and newer protocols, such as Blocks Extensible Exchange Protocol (BEEP).

2.3.2 XML messaging

This layer is responsible for encoding messages in a common XML format so that messages can be understood at either end. Currently, this layer includes XML-RPC and SOAP.

2.3.3 Service description

This layer is responsible for describing the public interface to a specific web service. Currently, service description is handled via the Web Service Description Language (WSDL).

2.3.4 Service discovery

This layer is responsible for centralizing services into a common registry, and providing easy publish/find functionality. Currently, service discovery is handled via Universal Description, Discovery, and Integration (UDDI). As web services evolve, additional layers may be added, and additional technologies may be added to each layer. Figure 2 summarizes the current web service protocol stack.

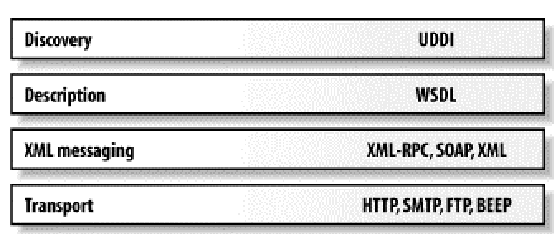


Fig. 2 Web Service Protocol Stack

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.4 Ontology

In the early 90's a couple of artificial intelligence based languages were developed. Knowledge representation (ontology) languages were first-order logic based like KIF [8] or Ontolingua [9].

In the last few years several ontology languages have been developed and many of them are well known in the context of the Semantic Web, especially those created by the World Wide Web Consortium (W3C). Such languages are commonly called web-based ontology languages or ontology mark-up languages. These languages are still in development phase, which means that they are continuously C3-2 evolving.

Most of them are based on the Extensible Mark-up Language (XML) syntax [7]. XML was specified as an open standard by the W3C to improve the information exchange via the Web. Despite the fact that XML was designed for the electronic processing of documents, it is widely used in a different range of application (i.e. for web services).

Therefore, the SHOE syntax was extended to use XML and later on, other ontology languages were built on the XML syntax as well. Other languages also have been used for building ontologies, like the Ontology Inference Layer (OIL) or (DARPA Agent Mark-up Language) DAML+OIL which was replaced by OWL.

Contrary to traditional ontology languages the Resource Description Framework (RDF) and RDF Schema are markup ontology languages. OWL is built on the top of RDF(S), which is the union of RDF and RDF Schema.

The stack of ontology mark-up languages and the relationships among them are shown in Figure 3.

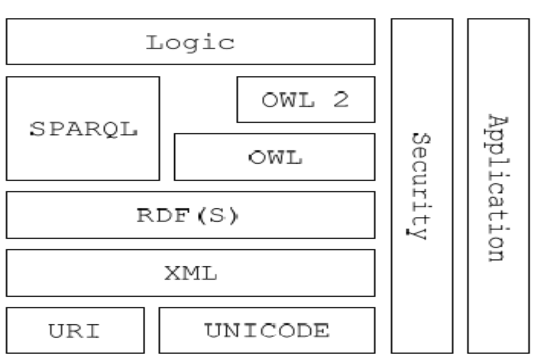


Fig. 3 stack of ontology mark-up languages

3. RELATED WORK

Figure 4 shows the various steps involved in approach for semantic based service discovery. Service categorization module will take care of service selection process for each process request.

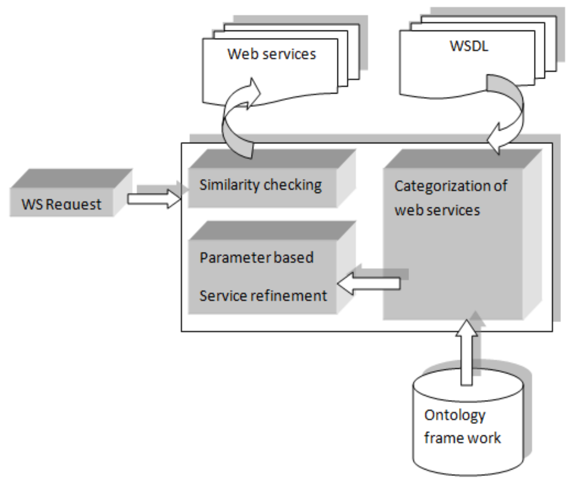


Fig. 4 Service categorization module.

Ontology can be used for web service categorization. In this approach web services are represented as a vector that comprises of the terms of the WS description and of the services' i/o parameters, it is called as service description vector(SDV) and is shown in the figure 5.

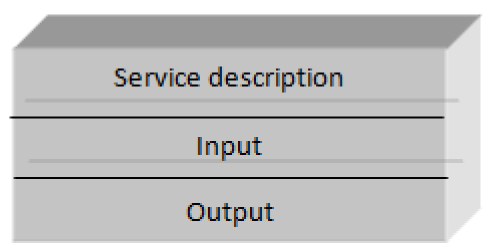


Fig.5 SDV

Improving the quality of the SDV categorization of WS is followed by the service selection from the relevant group of services is achieved by the parameter based service refinement. In which the relationships between WS input and output parameters may be represented as statistical association.

3.1 Procedures for integrated model

1. Process the service request and determine the overall search category of WS for the search

Journal of Theoretical and Applied Information Technology
15 June 2012. Vol. 40 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. Index the WS description collection and retrieve relevant descriptions

3. Process the services descriptions set and retrieve associated concepts related to the initial service request from the ontology framework.

4. Determine the similarity of the SWS using the frame work

5. Cluster the SWS and repeat the step 1 to 5

Ontology-based semantic Web services provide the basis for calculating the similarity between each other.

SDV can be used for calculating similarity of Let us consider the web services SWS_1 , SWS_2 and their SDVs. The ontology-based similarity of Web services which is presented in the document [10,11].

3.2 Similarity checking of the semantic web service

Algorithm *Checksimilarity(List_of_webservices)*
begin
1. Get List_of_webservice{ < SWS_1 , SDV_1> , < SWS_2 , SDV_2> .... < SWS_n , SDV_n> }
   - For each tuple do the following process
     If (SWS_1.name= SWS_2.name) then
     If(SWS_1.Description=SWS_2.Description) then

begin
For each input(I_i)of SWS_1.inputs
For each input(I_j)of SWS_2.inputs
Compare_Input(I_i , I_j)
For each Output(O_i)of SWS_1.outputs
For each Output(o_j)of SWS_2.outputs
Compare_output(o_i , o_j)
Add the similar web services into a group
end
end

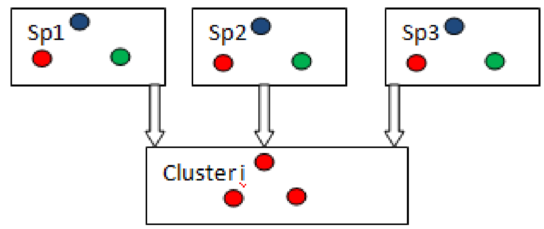
3.3 Clustering web services

The *Checksimilarity* procedure will produce the data according to their similarity. The *ClusterWS* procedure can be used to integrate the SWS.

Let us consider that the required no of cluter is tow. It can be selected from the outcome of the procedure *Checksimilarity*

Algorithm *ClusterWS (List_of_Similar_WS)*

begin
1. *For each SWS from SWS_GROUP*
   Cluster{ SWS_i, SWS_m }
   Next
end



4. CONCLUSION

In this paper i have proposed an integrated model which is based on semantic categorization of WS published in the UDDI. It dealt with selection of WS for a given service request and the refinement of WS based on input and output. Based on the relevant ontology terms WS request and the matching is enhanced.

As future work, i would like to extend the model to compare and integrate ontolgies. Hence, service provision is a crucial topic, which has a high impact on many different areas of research. Our main ambition of research is to enforce the service provision, since it is very likely that filling user requests on-demand will be common in the near future.

5. REFERENCES.

[1]. R. Akkiraju, J. Farrell, J.Miller, M. Nagarajan, M. Schmidt,A. Sheth, K. Verma, *"Web Service Semantics - WSDL-S"*, A joint UGA-IBM Technical Note,version1.0,April18,2005. http://lsdis.cs.uga.edu/projects/METEOR-S/WSDL-S.

[2]. http://www.uddi.org/specification.html

[3] M. K. Smith, C. Welty, and D. L. McGuinness. (2004). OWL Web Ontology Language Guide. W3 C Recommendation. [Online]. Available: http://www.w3.org/ TR/owl-guide/

[4]. Data communication and networking Behrouz A. Forouzan Third Edition.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[5]. W. Damm and D. Harel. LSCs: Breathing Life into Message Sequence Charts.In Proceedings of the IFIP TC6/WG6.1 Third International Conference on For-mal Methods for Open Object-Based Distributed Systems (FMOODS), page 451.Kluwer, B.V., 1999.

[6]. V. Haarslev and R. M¨oller. RACER User's Guide and Reference Manual Version1.7.

7. http://en.wikipedia.org/wiki/XML

[8].Genesereth, M., R. Fikes, 1992, Knowledge interchange format version 3.0 reference manual, Computer Science Department, Stanford University.

[9].Ontolingua, 1998. Ontolingua System Reference Manual, Knowledge Systems Lab, Stanford University.

[10] He Chaobo, Chen Qimai. *Fast approach for semantic webservice discovery [J]*. Computer Engineering and Design,2010, (12).

[11] Xu Dezhi, Zheng Chunhui, K. Passi. *Concept semanticsimilarity research based on SUMO [J]*. Journal of Computer Applications,2006, (01).