

# Web Services Discovery Approach: A Comparison Study

Shahid Khan<sup>1</sup> Nimble John<sup>2</sup> Govinda K<sup>3</sup>

<sup>1, 2, 3</sup> School of Computer Science and Engineering

<sup>1, 2, 3</sup> VIT University, Vellore - 632014, Tamil Nadu, India

*Abstract*---Web services are playing an important role in organization to improve their business. As web service applications are platform independent so large scale distributed systems can be developed easily using web services. As per organization requirements it is very difficult task to finding most suitable web service from large collection of web services for successful execution of applications. Keyword based web service discovery approach using UDDI is used in past. Apart from that many other approaches for discovering web services are also available i.e. syntax based and semantic based. To develop a system for service discovery which can work automatically is also the concern of service discovery approaches; it is also a complex task. In this paper, we give the comparison of different approaches for web service discovery. We present a survey of how these approaches differ from each other and find its pros and cons.

## I. INTRODUCTION

A web service is any application service that is available over the Internet, based on standardized XML messaging system. Web services can be used by any application and run on any platform i.e. provide interoperability. Web service description is provided by WSDL document. It is accessed through SOAP protocol. The discovery of web services plays a very important role. It must require the correct input, output, preconditions and effects specified by the user. And after that, quality parameters also need to have best web service from available web services. Web services provided by different service providers and published on internet using UDDI. UDDI is the mechanism for registering and discovering web services. It is platform independent registry as it is based on extensible mark-up language. It allows businesses to give list of services and describe how they interact with each other.

Nowadays, Web Services are developed using WSDL, RESTful based web Service and many traditional Services on Internet. Developer only focused on creating and publishing of web services. After Developed and published web services, discovery of those services is in main concern and many questions and researches are going on how to fetch exact web services and how to automate it as per requesters needs. At the time of fetching web services we must have to consider accuracy, efficiency and security factors because, number of available web services providing large functionality of same kind and it generates redundancies.

## II. LITERATURE SURVEY

Traditionally keyword based search approach using UDDI used for web service discovery. Discovery of the corresponding Web services is primarily semantic in nature at present scenario. But lack of semantics is the result of the

current keyword based representations that cannot express the correct and accurate meaningful services operate and the relationships among various entities in that services. We did the survey of different web service discovery methods that helps to recognize better methods for which applications [1].

Two Main Methods are WSDL based and Ontology based and further classifications are as follows:

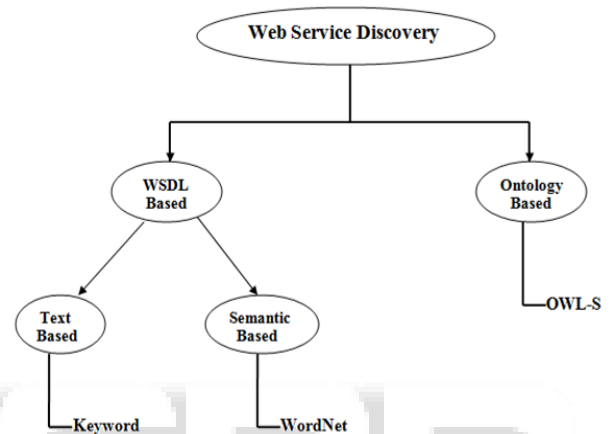


Fig. 1: Classification of Web Discovery Techniques

## III. TEXT-BASED WEB SERVICES DISCOVERY

It is generally simple approach to perform Web service discovery. The most commonly used text-based method is **keyword based discovery** build in the UDDI public registry. The UDDI API permits developers to give keywords of particular services that they want and then returns a list of web services whose service description contains those keywords. The major advantage is that easily extent to a large number of services and can utilize established keyword matching technologies. Disadvantages of keyword based techniques are imperfect due to the ambiguities of natural language and the lack of semantics [1].

## IV. SEMANTICS BASED WEB SERVICES DISCOVERY

Present Web services only attention on technical principles which allow users to exchange information in a standardized manner. They not concentrate on semantics of web services and web service descriptions. The lack of semantics in description forms inefficiencies in utilizing the Web service discovery. If web service describe with semantics then it provides the ability for automatic Web service discovery, easy invocation, interoperation and composition, and Web service execution checking.

This method divides into two parts: -

- 1) Service Publishing and
- 2) Service Discovery

First, Web service providers explain the terms and ideas in their web services and their interrelationships to establish the semantics for describing facilities of their services. This is completed by creating an ontology document, it is a documents or a file that properly defines relations between expressions or keywords with some meanings [2].

For example, if a hotel wants to would publish its hotel services in UDDI registry, it first describe the hotel domain in an ontology with domain classes such as room booking, hotel location, user identity, booking confirmation number, credit information, booking start date, and duration. And the next step, describing a detail discussion of the semantic relationship for service description.

And next Service providers explain their services with semantic information. It contains information about the functional attributes of a service and the properties of the service namely the inputs, outputs, preconditions and effects. The service discovery module receives requests and executes the query. Service type filter is applied and it performs a UDDI group-based search to retrieve all those services that fall under the specified set of groups in given taxonomy. These filtered set of services are then passed into semantic matching engine and it matches the inputs, and outputs of a service request with those of a service. Two conditions are follow first if match exactly, that can be inference from the ontology using an inference engine. If no direct matches found, our semantic matching engine automatically finds ways in which two or more services could be composed to meet the original request [2].

## V. WORDNET

It is also a method of semantic web and it based on words. In past we used dictionary organized by the alphabetical order and has a strong power to query words, but can't provide the relation, such as synonymy, antonyms etc. As coming of semantic dictionary meets the requirements of intelligent application and widely used in computational linguistics and natural language processing. A WordNet is used; it is online database of huge words and their semantics WordNet groups words together based on their meanings. And divides words into nouns, verbs, adjectives, adverbs. These four groups are organized in different ways. WordNet organizes the nouns depending on the hierarchy of subjects, verbs according to lexical collection relationship, adjectives and adverbs according to N-dimensional space [3].

### A. Ontology based Web Services Discovery

Currently web service's implementation and execution has been focused on the syntactical details of the web service. This implementation limits the web service to keyword based search. In order to remove inefficient keyword based search, represent information with metadata help us for efficient search. Adding the semantic information with syntactical web service it helps the purpose and usage of web service. Ontology provides the description of web services. Ontology is used basically in semantic web for understanding of particular concept. Several operations are implemented by the web services. To describe it, the behaviour aspect is used. Web service operation has input and output. Ontology captures the more real world information knowledge but we use minimum information as

an output. Detail information can be represented by the WSDL or UDDI. There have been many advances on ontology to help resolve the goal of sharing knowledge for various domains of interest [2].

### B. OWL-S

It stands for Ontology Web Language with Semantics. It is based on OWL with some refinement and adds additional information and properties. OWL-S supports automation of service selection, translation of message content between various services, invocation and service composition.

OWL-S is divided into three elements: Profile, Process and Grounding

Profile describes capabilities of Web Services as well as extra characteristics (e.g. inputs, outputs, preconditions and effects) of web services thus it is key in the web service discovery process.

Process provides an explanation of the activity of the Web Service provider from which the Web Service requester can get the interaction

Grounding is an explanation of how abstract information exchanges expressed in the process model are mapped against actual messages that the provider and the requester exchange. Take advantages of the semantic matching and the capability that can be embed OWL-S Profile information inside UDDI advertisements for increasing the discovery approach.

After OWL-S information submitted in the UDDI advertisements and queries, we can also extend the UDDI registry and the UDDI API used to access it. It used hierarchical or relational model for extending i.e. we consider an example of living things and it can be both human and animals and further classifies [2].

The UDDI API is extended with new classes to represent semantic queries and results.

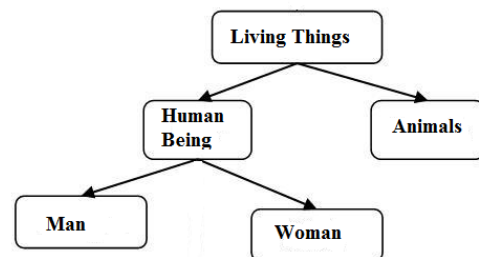


Fig. 2: Basic Ontology Classes and Relations

In this example we apply the OWL-S Modules i.e. Profile, Process and Grounding. In Profile Module refine the outer properties of living things and some extra features. And secondly Process describes the all properties of living things and their relations and classifications. And finally the Grounding module enhances the abstract details of process module and relate with service request requirements. Thus the OWL-S has more benefits compare than OWL and it automate the service discovery and enhance service accessibility.

Some advantages of OWL-S are as follows: first is the functional level of OWL-S does not depend on any taxonomical representation. And the second is the OWL-S

profile is represents explicitly data that the service manipulates [4].

Disadvantages of OWL-S is that solves only a particular domain knowledge and not suitable for deep reasoning.

## VI. CONCLUSION

In this paper, we comprise given outline of different web service discovery approaches with their advantages and disadvantages. Different methods have different way to discover the given web services. Some techniques are taking into consideration the concept of basic traditional text based keyword matching mechanisms, and other focus on semantic web methods. Keyword based mechanism of discovery had lots of drawbacks and can't be automated. Presently semantic based mechanism like RDF, RDFS, OWL, OWL-S are very popular and it helps to improve the performance of discover the web services and automate it. This is the solutions for automatic discovery are taking more attention. So our paper helps the developer and researchers to find out which is best techniques for web discovery in present scenario.

## REFERENCES

- [1] Searching services "on the Web": A public Web services discovery approach , Image technologies and Internet-Based System 978-0-7695-3122-9/08 \$25.00 © 2008 IEEE DOI 10.1109/SITIS.2007.17327 Third International IEEE Conference on Signal-Image
- [2] Semantics Based Web Services Discovery, Science and Technology Commission Foundation of Shanghai, China under contract 05DZ11C06
- [3] Web Service Matching based on Semantic Classification, School of Health and Society Department Design and Computer Science Kristianstad University SE-291 88Kristianstad Sweden
- [4] [http://www.knexusresearch.com/publications/pdfs/Case based%20classification%20alternatives%20to%20ontologies%20for%20automated%20web%20service%20discovery%20and%20integration.pdf](http://www.knexusresearch.com/publications/pdfs/Case%20based%20classification%20alternatives%20to%20ontologies%20for%20automated%20web%20service%20discovery%20and%20integration.pdf)
- [5] <http://www.di.ufpe.br/~redis/intranet/bibliography/middleware/chen-composition06.pdf>