A Survey on Semantic Web Service Discovery

[1] Malaimalavathani. M, [2] Gowri. R
[1] PG Student, Department of CSE, Sri Manakula Vinayagar Engineering College,
[2] Associate Professor, Department of IT, Sri Manakula Vinayagar Engineering College, Puducherry,
Email: vathani.ani@gmail.com, gowrithirugnanam@gmail.com
Ph: 9626960199, 9600973073

***Abstract*— Web services enhance current web functionality by altering its nature from document to service oriented. These services are self-describing, self-contained, modular applications accessible over Internet. Web service technology has gained more important role in developing distributed applications and systems on the Internet. Rapid growth of published Web services makes their discovery more and more difficult. Service discovery is one of challenging issues in semantic web services. A Semantic-based approach for WS discovery searches based on both functional and non-functional parameters and also recognizes the importance of interoperability with the Web services. Currently, most of the existing service discovering and matching approaches are based on keywords-based strategy. More services are available; the most is it becomes difficult to find the most appropriate service for a specific application. In this paper we mainly discuss about evaluating the discoverability of services through semantic web services.**

***Keywords:* web services, WSDL, UDDI, services discoverability, semantic web services.**

I. Introduction

Web Services (WSs) are modular, self-describing, and loosely coupled software applications that can be advertised, located, and used across the Internet using a set of standards such as SOAP, WSDL, and UDDI. Web services, as a key technology for realizing service-oriented architectures, promise to enable interoperability and integration between heterogeneous systems and applications. The discovery and selection of the appropriate services to fulfill a given request constitutes a fundamental task in such architectures. However, current industry standards for registering and locating Web services (WSDL, UDDI) aim at describing the structure of the service interface and of the exchanged messages, limiting the discovery process to essentially keyword-based search. Even though interoperability at the syntactic level is a necessary requirement, the identification and selection of appropriate services should be done in terms of the semantics of the requested and offered capabilities. To this direction, the Semantic Web, through the use of ontologies, provides the means to enrich the service descriptions with semantic information, allowing software agents to reason about the terms in these descriptions. This is a significant step for increasing the precision of the discovery process, as well as for minimizing the required human intervention. Several approaches have been proposed for adding semantics to Web service descriptions, including OWL-S, WSDL-S, and WSMO.

Efficiently finding Web services on the Web is a challenging issue in service-oriented computing. Currently, UDDI is a standard for publishing and discovery of Web services, and UDDI registries also provide keyword searches for Web services. With the development of Semantic Web technologies, more and more Semantic Web data is generated, which is being used in Web applications and enterprise information systems. To effectively utilize the large amount of semantic data, efficient search mechanisms customized for Semantic Web data, especially for ontologies, have been proposed for both humans and software agents.

Web services should be semantically annotated to provide the best match to the service requestor as per his requirements. In order to address these problems, an efficient Web services on the Web based on their associated semantics is presented in this paper. The main objective is to develop an effective mechanism for Web service discovery.

Web Service Model: In a Web service model, Web services which provide functions or business operations which can be deployed over the Internet. Publishing, binding, and discovering Web services are three major tasks in the model. Discovery is the process of finding Web services provider locations which satisfy specific requirements of the customer. Web services are useless if they cannot be discovered. So, discovery is the most important task in the Web service model. The interaction between a service requester, service providers, and a service discovery system are as follows:

1. The service providers are created by companies or organizations. In order to be invoked, the Web services must be described. This will facilitate discovery and composition. WSDL or service profile of semantic Web service is used to carry out this function.

2. The Web service requester describes requirements in order to locate service providers. Service requesters usually contain a description of the Web service, though it is not a Web service which can run on the Internet.

3. The Web service discovery or service registry is a broker that provides the search function. The service providers advertise their service information in the discovery system. This information will be stored in the registry and will be searched when there is a request from service requester. UDDI is used as a registry for Web service.

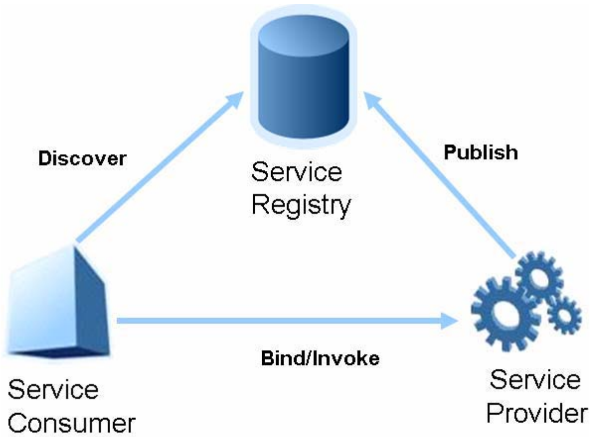

Figure 1: Web Service Model

Publish: The Web service providers publish their service information through the discovery system for requesters to discover.

Discovery: The Web service requesters retrieve service providers from the service registry. Based on service descriptions, which describes the requirements of the Web service requesters, the discovery system will output a list of Web service providers which satisfy the requirements.

Bind: After discovering, the discovery system provides a number of Web service providers. The Web service requester invokes these Web service providers. The binding occurs at runtime.

The Web service requesters and Web service providers will communicate via SOAP protocol for Web service exchange information.

II. Service Discovery

Web service discovery is a process of discovering service that most suitable to user's request according to requester's requirement. Discovery is one of the major challenges of the web service technology. An effective and automated search and selection of relevant services is necessary for both human users and programs.

Steps performed to make service discoverable:

- Making service discoverable by adding interpretable meta-data
- Document the information about the service in a consistent manner
- Store the documented information in a searchable repository
- Enable others to search for the documented information in an efficient manner.

Service discovery standards:

WSDL and *UDDI* are two standards used in service discovery. The Simple Object Access Protocol (SOAP) is a messaging protocol used to invoke web services and get back results asynchronously.

Web Services Description Language (WSDL):

WSDL is a language for description of service and contains operations supported by the service. Each operation is described by it's input and outputs. *WSDL* description of a service defines XML message format for communication with the service. A compiler at the client generates stubs based on the WSDL description for:
1) Marshaling and unmarshaling objects into SOAP messages.
2) Sending SOAP messages over communication protocol.
The application is then bound with these stubs to invoke the service.

Universal Description and Discovery Interface:

UDDI is a registry that contains information about different services offered by various service providers. This information is usually output as a WSDL document. UDDI provides mechanisms for:
1) Publishing a service to the registry
2) Searching a required service from the registry.
The state of the art today limits storage in UDDI and searches are syntactic in nature.

Ontology:

In Semantic Web research, ontologies provide the foundation for machine-processable data and allow to exchange information between people and machines by both syntactic and semantic. Ontology represents knowledge about a particular domain. This knowledge includes entities in the domain, their property and relationship with each other. Entities in the ontology are termed *Concepts*. A well defined syntax is required to represent concepts of a domain. RDF framework is suitable for describing ontology. Web Ontology Language (OWL) is developed on the top of RDF and is used for ontology description.

III. Semantic Web Services

Semantic Web Services like conventional web services, are the server end of a client–server system for machine-to-machine interaction via the World Wide Web. Semantic services are a component of the semantic web because they use markup which makes data machine-readable in a detailed and sophisticated way. The Semantic Web adds metadata and ontology information to Web pages to make the Web easier to exploit by both humans and especially for programs. The paradigm of the Semantic Web helps use metadata as a largely untapped source in order to enhance activities of intelligent information management.

Advantage:

As the name indicates-stem from the fact that semantics is encoded explicitly and formally in the service description and that the execution environment for semantic web services is able to interpret and use the semantics appropriately.

Semantic Web Service Description Languages:

Web service discovery by utilizing the machine readable constructs of the representation. Several standards have been proposed for creating semantic Web services. Each one of them is having their own strength and can be used in a specific situation. Some of the popular languages are described as follows.

A Semantic Web Services platform that uses OWL (Web Ontology Language) to allow data and service providers to semantically describe their resources.

OWL-S:

OWL-S is an OWL-based Web service ontology, which supplies Web service providers with markup language, constructs for describing the properties, and capabilities of their Web services in unambiguous and computer interpretable form. An OWL-S description is composed of three parts which are Service Profile, Service Model, and Service Grounding.

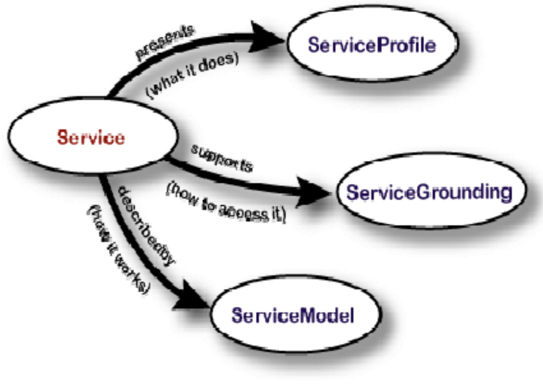

Figure 2: Upper level OWL-S Service Ontology

Service profile: Presents “what the service does” with necessary functional information: input, output, preconditions, and the effect of the service. It is used for advertising and discovering services.

Service model: Describes "how the service works”, that is all the processes the service is composed of, how these processes are executed, and under which conditions they are executed. It gives a detailed description of a service's operation.

Service grounding: Describes “How is it used”. It provides details on how to interoperate with a service, via messages.

WSDL-S: Current WSDL standard operates at the syntactic level and lacks the semantic to represent the requirements and capabilities of Web Services. WSDL-S is a lightweight approach for adding semantics to Web services. In WSDL-S, the semantic models are maintained outside of WSDL documents and are referenced from the WSDL document via WSDL extensibility elements.

WSMO: WSMO provides a conceptual framework and a formal language to describe all relevant aspects of Web services to facilitate the automation of service discovery using semantics. The overall structure of WSMO is divided into four main elements.

(i) Ontologies: provides the terminology used by other WSMO elements.
(ii) Web service descriptions: describes the functional and behavioral aspects of a Web service.
(iii) Goals: represents user desires.
(iv) Mediators: aims to automatically handle interoperability problems between different WSMO elements.

OWLS-SLR: (SLR stands for Structural and Logic-based Reasoning) is a semantic Web service matchmaker written in Java that makes use of OWL-S Profiles. It follows the abstract Web service discovery approach, performing matchmaking based on inputs, outputs and non-functional properties. Some of the main features of OWLS-SLR include:

A. *Taxonomy-based matchmaking:*

OWLS-SLR allows the existence of a Profile Taxonomy, that is, a subclass hierarchy of the OWL-S Profile concept. This is feasible by performing complete ABox DL reasoning on OWL-S advertisements and queries that are both represented as direct or indirect Profile ontology instances. OWLS-SLR has been defined on top of the Pellet DL reasoner.

B. *Structural ontological knowledge:*

OWLS-SLR implements two concept distance measures, namely the *edge distance* and the *upwards cotopic* distance. The former computes the distance of two ontology concepts based on the number of edges found on their shortest path. The latter determines the similarity of two ontology concepts taking into account the ratio of the common superclasses of the two concepts. In that way, we are able to match concepts not based only on logic-based DL reasoning, but also on the structural knowledge that the ontology provides, returning useful matchmaking results.

C. *Role-oriented matchmaking*:

OWLS-SLR allows the annotation of the input and output parameters of the Profile concept with ontology roles. This is achieved by extending the Input and Output OWL-S concepts and exploiting the excellent classification capabilities of DL reasoning that is based on anonymous concept definition. The rationale behind this direction is to allow the ontology roles to play an important role during matchmaking, since they encapsulate important domain knowledge.

D. Filtering:

The functionality of OWLS-SLR is characterized by a number of filters that allow users to control the matchmaking results according to the application domain and their preferences.

IV. CONCLUSION

In this paper we discussed about the development of Semantic Web technologies. The semantic Web service technology aims to describe the Web services in an unambiguous and machine-processable form for enabling the intelligent interoperation of Web services. Ontology, which provides a formal and explicit specification of a shared conceptualization, has been expected to enrich Web services with semantics. A Semantic-based approach for WS discovery searches based on both functional and non-functional parameters and also recognizes the importance of interoperability with the Web services. Although current standards like UDDI, WSDL, and SOAP form the basis of making Web services a workable and broadly adopted technology they are syntactic and therefore are plagued with problems like inefficient service discovery and composition. Semantic Web technologies are likely to provide better qualitative and scalable solutions to these problems.

REFERENCE

[1] "Structural and Role-Oriented Web Service Discovery with Taxonomies in OWL-S", Georgios Meditskos and Nick Bassiliades, Member, IEEE Transactions On Knowledge And Data Engineering, Vol. 22, NO. 2, February 2010

[2] "Semantic Web Services", Sheila A. McIlraith, Tran Cao Son, and Honglei Zeng, *Stanford University,* IEEE Intelligent Systems 2001

[3] Guo Wen-yue, Qu Hai-cheng, Chen Hong, "Semantic web service discovery algorithm and its application on the intelligent automotive manufacturing system", International Conference on Information Management and Engineering, IEEEXplore, 2010.

[4] Bo Zhou, Tinglei Huan, Jie Liu, Meizhou Shen, "Using Inverted Indexing to Semantic WEB Service Discovery Search Model", 5th International Conference on Wireless Communications, Networking and Mobile computing, IEEEXplore, 2009

[5] Berners-Lee, T. Hendler, J., Lassila, O.: "The Semantic Web". Scientific American, Vol. 284 (4). (2001) 34-43

[6] "OWL-S: Semantic Markup for Web Services". http://www.w3.org/Submission/OWL-S/.

[7] "Web Ontology Language (OWL) Guide". http://www.w3.org/TR/2002/WD-owl-guide 20021104