Semantic Web Service Description, Discovery and Integration using Schema Matching Method

Neelendra Badal, Divya

***Abstract*— as the World Wide Web has grown, the next step of World Wide Web is web services. Web is the collection of services that interoperate with the help of internet. All the programmable materials are to be placed in the web sites which are accessed by others with the help of distributed nature of the network. The Universal Description, Discovery and Integration are used to describe, discover and integrate the relevant web services. Universal Description, Discovery and Integration are the distributed web-based information registries of web services.**

***The information's are described in the various domain Ontology and Generic Procedures. Web is alone not sufficient to describe the data because single web is not capable to carry all the information. Here, the word "Semantic Web" plays important role due to its property that is a combination of number of services. In this work Services are described via Web Service Description Language, Resource Description Framework and Ontology Web Language etc. in Semantic Web for better results.**

***Further, the described Services in Knowledge bases are discovered with the help of schema matching algorithm. In this work schema matching algorithm is based on various different similarity measures. The modified architecture for the discovery of web services has also been presented in this work.**

After the description and discovery of web services, the integration of these services is required with the services offered by different-different business. The Composition methods are used for integrating these web services in this presented work. When the required services are discovered services are automatically integrated and provide the services to the customers. In this work semantic schema method has been used for the description, discovery and integration of web services. The new discovered services are also kept in the knowledge bases for the further future consideration. An experimental result has been also presented in this work which also shows that semantic schema matching is better than simple UDDI.

Keywords –

I. INTRODUCTION

It is important to conduct faster and more efficient so, a web service provides sites in which there is exchange information on demand. Automatic discovery and integration of services arises challenges: first as there are large amount of web services to find out the locations of suitable service that provide a solution. Second once the services should be discovered after that able to integrate with those services automatically.

Both the challenges are related to the ability how the services should be matched. The webs are represented in HTTP and HTML. HTTP is defined how data should be transported between server and client and HTML defines the data format. But these technologies are not sufficient for meaningful communication between the parties.

With the help of semantic web described in [6] the concept of current web should be enhanced where properties, capabilities, interfaces, and effects are encoded in a machine interpretable form with clear meaning. Services can be discovered with the help of web services which are semantic in nature. Semantic web is developed by using the technologies named as eXtensible Markup Language (XML) [15] and Resource Description Framework (RDF) [9].The limitation of XML is their lacks of an explicit semantics that mean two identical XML represents the two different meaning depends who and when use them. DAML-S is markup language that provides a logically grounded view of the web services. SOAP and WSDL in [5] and [3] are designed to provide descriptions of message transport mechanism and interface used by each service.

The semantic representation of web services should be represents effectively by the support of DAML-S which is tightly connected with DAML+OIL. In [7] the DAML+OIL represent the definition of relations between concepts.

In this our goal is to efficiently describe the web services, discovered and integrate with the help of modified architecture of UDDI described in [4] and [1]. Finally present conclusion and future direction of this work.

II. PROBLEMS WITH UDDI

UDDI is designed for industry initiative for enhancing business. UDDI contains directory service for web services offered by businesses as shown in [2]. In these directories the various businesses can register themselves. The UDDI directory contains the whole information about the web services like about business entities, business nature and how to access them. These services are use with the help of WSDL and SOAP.

The main aim of UDDI is the fastest discovery of the web services requested by the customers. The web services can be searched by name, business, location and t-Models. Each service may have more than one t-Model. There are two types of t-Models supported by UDDI.

I. That expresses the technical specification of the services (e.g. protocol, interchange format etc).
II. That expresses abstract specification of the services and taxonomies schemes.

As know UDDI works in three phases: description, discovery and integration. The UDDI directories give the whole specification of the web services offered by the business. The discovery is depends on the directories of the UDDI. There are three major limitations are associated with UDDI. These limitations are as:

Manuscript received February 15, 2014.

Neelendra Badal is an Assistant Professor in the Department of Computer Science & Engineering at Kamla Nehru Institute of Technology, (KNIT), at Sultanpur (U.P.), India.

Divya is an Assistant Professor in the Department of Computer Science & Engineering at Feroze Gandhi Institute of Engg. & Technology, (FGIET), at Raebareli (U.P.), India.

I. It is not capable of finding the entities relationship store in the directories.
II. UDDI's discovery is based on the high level information described about business and their services.
III. It follows only the direct matches, if there is no direct matches then it is fail to find out the respective services else the indirect matches can be fulfill the requests.

By enhancing the concept of semantic web as shown in [11] is tried to cope with these limitations to make UDDI should be capable for discovery of web services.

After the successful discovery of web services the results are compose for the integration of services requested by the customers.

III. HOW SEMANTIC MATCHING DONE WITH UDDI

As shown in Fig. 1 the service provider sends their services in the form of advertisements to the communication module. The communication module sends this request to the DAML-S/UDDI translator.

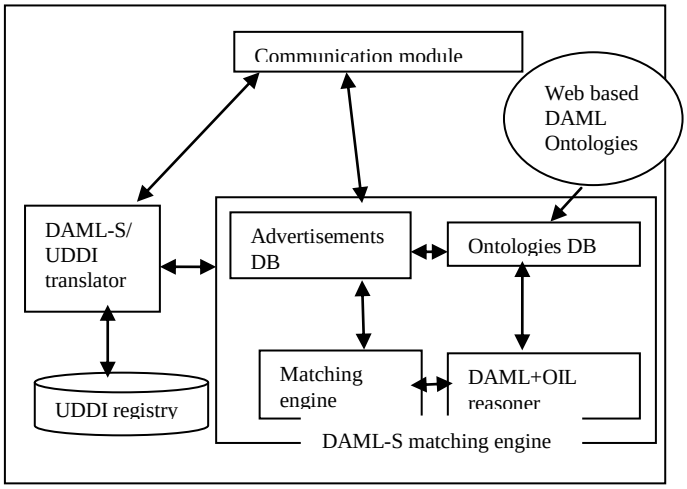

Fig. 1: Architecture of DAML-S/UDDI Matchmaker

In [14] UDDI register their services using service provider and service name. As a result a unique ID is generated for the particular service.

This ID with the Particular advertisement is forward towards DAML-S matching engine for the semantic matching which is described in [6].

If any customer requests for any service then this model works in opposite direction as the communication module forward the request towards DAML-S matching engine where the request should be match with the advertisements from the advertisements database that best fit for the current request.

The matching is computed with the use of DAML + OIL reasoner. It uses the Ontology database for data for computing the matching. Due to this the discovery process is the matching is computed with the use of DAML+OIL reasoner. It uses the Ontology database for data to compute the matching. Due to this the discovery process is fast. The results are forward towards the UDDI record. The combination of the UDDI record and advertisement are forward to the requestor.

IV. ARCHITECTURE OF UDDI WITH SEMANTICSCHEMA MATCHING

Service providers describe the business, their entities, interrelationship with the help of UDDI registries maintained. The registries are maintained by either creating an Ontology document or select Ontologies which will be suitable for the particular from the existing repository of Ontology. Ontology contains the information about the relation between the terms. The relationship among the terms is represented in well defined language. The language used to represent the Ontology is DAML + OIL. After that service provider announce their services with semantic information in DAML-S.

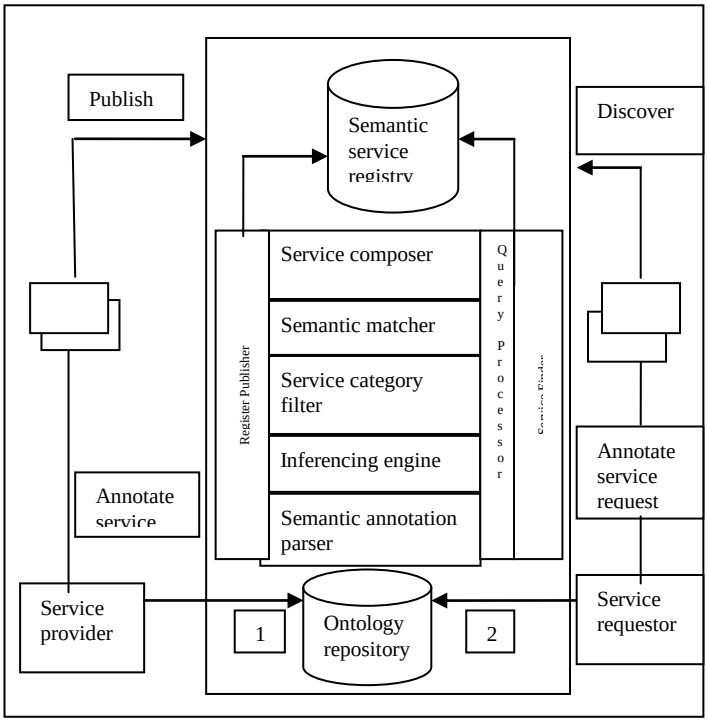

Fig2: Architecture of Semantic Schema Matching

This contains the whole information about the service providers like as properties (inputs, outputs, preconditions and effects) and functional attributes (quality, geographical status etc).

Overall it defines the capability of the service described in [1]. The announced services are published in registry of UDDI. The service category module is used to capture the UDDI taxonomy information with the help of this the publish registry published the specified services under UDDI taxonomy. The main work of this module is to convert the DAML-S services to UDDI records. After that publishes the service under the specified taxonomy. As a resultant the semantic information about a service provider is mapped with meta information which is associated with a UDDI businessEntity data structure as in [13]. The properties of service are represents with the help of t-Model created from DAML-S descriptions.

As shown in figure 2. The module service finder is responsible for receives the requests and perform on inquiry. If the related inquiry is in UDDI then a module service category filter receives all the services that satisfy the given request. This is done by using UDDI find method. The services

that are filtered are passing towards to the semantic matcher.

The matching module matches the inputs and outputs of a service required with those of a service as described in [10]. There are two possible cases for matching:

I. Either match exactly or

II. By some relationship that can be inference from the Ontology using an inferencing engine.

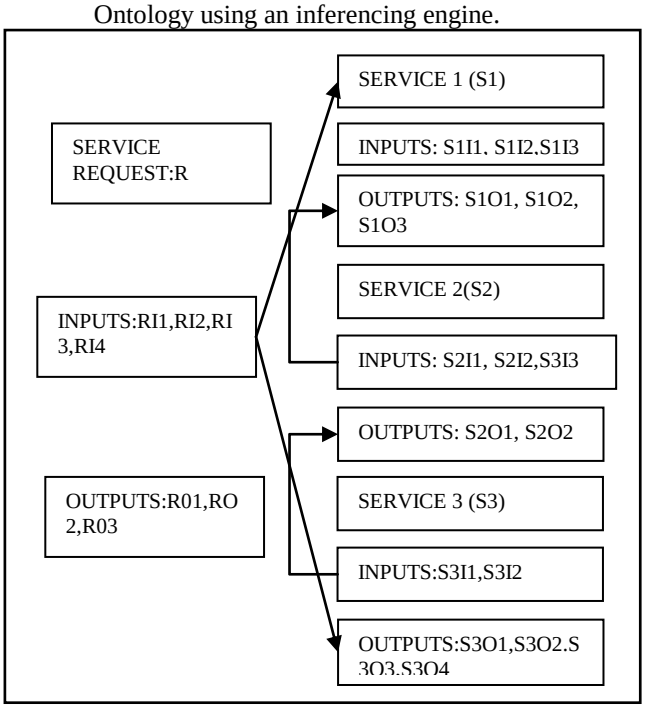

Fig: 3. backward chainer

Exact match for the requested service is preferred. It is possible that more than one input in the input sets of the request and the service. For the perfect matching the matching engine consider the maximum of the match distances between the corresponding input properties and the service to determine if the input are a match as described in [4]. Same procedure is applied to match the output. When all the results should discover the resultant should be composed as discovered with the help of [8].

V. EXPERIMENTAL RESULTS

```
public class TextMatching
  {    public TextMatching()
    {
    }
 public static int ComputeEditDistance(string s,
string t)
        {
            int n = s.Length;
            int m = t.Length;
 int[,] distance = new int[n + 1, m + 1];
            int cost = 0;
if (n == 0) return m;
            if (m == 0) return n;
for (int i = 0; i <= n; distance[i, 0] = i++) ;
            for (int j = 0; j <= m; distance[0,
j] = j++) ;
      for (int i = 1; i <= n; i++)
            {
      for (int j = 1; j <= m; j++)
                {
cost = (t.Substring(j - 1, 1) == s.Substring(i -
1, 1) ? 0 : 1);
 // all cost op of 1
                    distance[i, j] =
MathLib.Min3(distance[i - 1, j] + 1,
                    distance[i, j - 1] + 1,
        distance[i - 1, j - 1] + cost);
```

Fig. 4: Module for Text Matching

The Fig. 3 has been presented here for the discovery of Web Services using class text matching.

VI. ANALYSIS AND GRAPHICAL RESULTS

The comparisons have been done with help of tool uses SHARP and SPNP with the help of PTIME as given in Table 5.2.

Table 5.2

Comparison between UDDI and S-UDDI

No. of jobs	UDDI (ms)	SUDDI (MS)	Performance improved in SUDDI (%)	Average Time taken by UDDI (ms)	Average Time Taken by S-UDDI (ms)	Performance increased by S-UDDI in %
2	5.234	4.44	84.83	70.164	59.19	84.35%
4	10.02	8.53	85.12			
6	13.44	11.07	82.26			
8	18.23	15.60	85.57			
10	23.24	19.55	84.12			

Using this table the performance of SUDDI is found out. The results show that Semantic UDDI enhances the performance of discovery up to 84.35%.

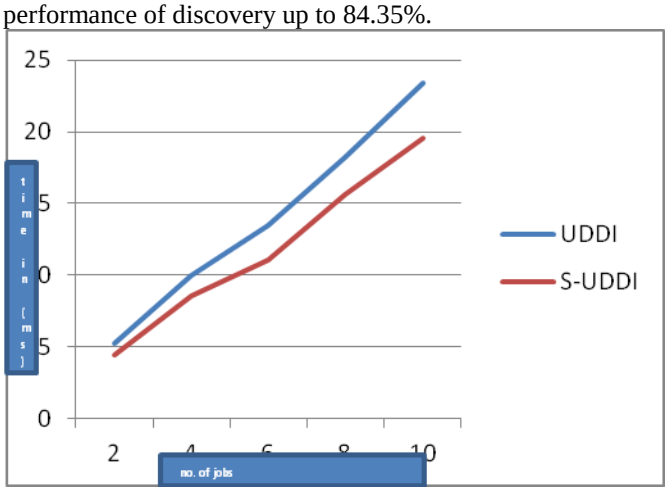

Fig. 5.4: Graphical Representation of UDDI and S-UDDI

VII. UDDI EXPLORER FOR SEARCH ENGINE

After the completion of Discovery process the results should be composed for completing the integration phase and as shown in Fig. 5.7 there are no results match the criteria

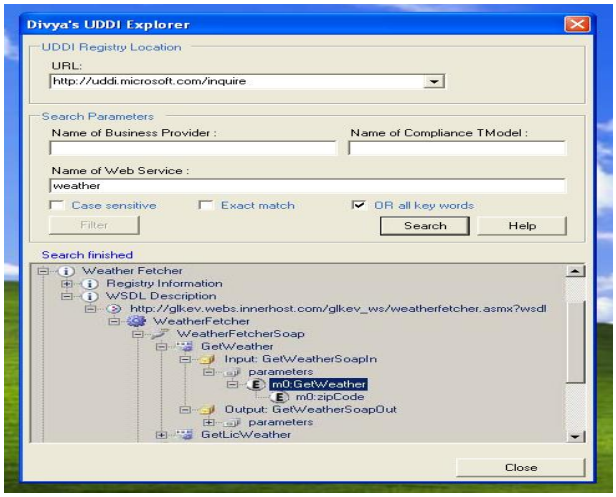

Fig.4: shows the discovery process

VIII. CONCLUSION AND FUTURE SCOPE

In this paper presented the description of the web services using directory service. It also enhances the discovery of web services in UDDI with the help of semantic web. The concept of traditional integration of the inquiry capabilities is not efficient. It is enhance by performing the automatic service composition using DAML-S semantics of services. With the help of this architecture all the three requirements of UDDI are performed efficiently and complete the cycle of description, discovery and integration. Service providers describe the in this dissertation the problem associated with UDDI and scope of Schema based UDDI has been discussed. Various technologies concerned with UDDI are reviewed and examined.

The modified architecture is implemented for the description of Web Services. With the help of Semantic Web the services are successfully described in the registries.

The desired information is discovered from the large databases with the help of Semantic matching mechanism. Through the architecture of discovery, web services are successfully discovered.

The results obtained from the description and discoveries are integrated to fulfill the user's requirements with help of various composition methods. UDDI Explorer tool is developed which explore the Semantic UDDI. Semantic UDDI is performed well in comparison to simple UDDI and enhances the results of discovery.

In future, Dynamic nature may be incorporated for the selection of URL as well as services provided by the selected URL. Further, performance may improve.

REFERENCES

1. Bernstein, P.A., Melnik, S., Petropoulos, M. and Quix, C., 2004. Industrial strength schema matching. SIGMOD record, Vol. 33(4), pp. 38-43.
2. Brown, A. and Haas, H. in 2002. Web Service Glossary. http://www.w3.org/TR/ws-gloss/>. World Wide Web Construm (W3C), HTML.
3. Christensen, E., Curbera, F., Meredith, G. and Weerawarna, S., 2001. Web services (WSDL). W3C site: http://www.w3.org/TR/2001/-NOTE-wsdl-20010315 [visited on September 2005].
4. Clemant, L., Hately, A., C.V and Rogers, T., 2004. UDDI spec Technical committee draft. UDDI web site http://uddi.-org/pubs/uddi-v3.0.2-20041019.htm [visited on September 2005].
5. Gudgin, M., in 2002. Simple Object Access Protocol (Soap). , HTML.http://www.w3.org/TR/SOAP/>, World Wide Web Cconstrum (W3C)
6. Massimo Paolucci, Takahiro Kawamura, Terry R. Payneand Katia Sycara. Semantic matching of web services capabilities in ISWC2002.
7. Mcguinness, D.l. and Harmelen, F.v., 2004. OWL Web Ontology Language overview. W3C web site: http://www.w3.org/-TR/owl-features/[visited on January, 2006].
8. PDDL Technical Committee. Pliannning Domain Definitionn Language in 1998. http://www.dur.ac,uk/d.p.long/IPC/pddl.html.
9. RDF Technical Committee. Resource Description Framework: RDF in 1999. http://www.w3.org/TR/1999/REC-rdf-syntax-19990222.
10. Tim Berners Lee, HendlerJ., Lassila. The Semantic Web. Scientific American, Vol. 5/01, May 2001.
11. Trastour, D., Bartolini, C. and Castillo, J.G., 2001. A semantic web approach to service description for match making of services. Pape presented at thr 1st Semantic Web Sumposium California, USA.
12. SOAP Technical Committee. Simple Object Access Protocol in 2000. http://www.w3.org/TR/SOAP.
13. UDDI Technical Committee. Universal Description, Discovery and Integration 2002.
14. UDDI Technical Committee. UDDI version 2.0.4 API Specification in 2002.http://www.oasis-open.org/committees/uddi-spec/tcspecs.shtml #uddiv2.
15. XML Technical Committee. Exte. Markup Language: XML in 2000. http://www.w3.org/TR/REC-xml.

AUTHORS PRFILE

Neelendra Badal, is an Assistant Professor in the Department of Computer Science & Engineering at Kamla Nehru Institute of Technology, (KNIT), at Sultanpur (U.P.), INDIA. He received B.E. (1997) from Bundelkhand Institute of Technology (BIET), Jhansi (U.P.), INDIA, in Computer Science & Engineering, M.E. (2001) in Communication, Control and Networking from Madhav Institute of Technology and Science (MITS), Gwalior (M.P.), INDIA and PhD (2009) in Computer Science & Engineering from Motilal Nehru National Institute of Technology (MNNIT), Allahabad (U.P.), INDIA. He is Chartered Engineering (CE) from Institution of Engineers (IE), India. He is a Life Member of IE, IETE, ISTE, CSI, India. He has published more than 35 papers in International/National Journals, conferences and seminars. His research interests are Distributed System, Parallel Processing, GIS, Data Warehouse & Data mining, Software engineering and Networking.

Divya, is an Assistant Professor in the Department of Computer Science & Engineering at Feroze Gandhi Institute of Engg. & Technology, (FGIET), at Raebareli (U.P.), INDIA. She received B. Tech. (2006) from Moradabad Institute of Technology (MIT), Moradabad (U.P.), INDIA, in Information-Technology, Persuing M.Tech. (2008) in Computer Science & Technology from Gautam Buddh Technical University, Lucknow (U.P.), INDIA. She has presented two papers in national conference . Her research interests are Distributed System, Computer Graphics and Networking.