IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 2, May 2012
ISSN (Online): 1694-0814
www.IJCSI.org

A Semantically Enhanced Searching Framework to Discover the Sensor Web Services for Wireless Sensor Network

Manoranjan Parhi [1], Chinmohan Nayak [2], Sanjaya Kumar Jena[3], Niranjan Panda[4], Pankajini Panda[5]

**[1] Dept. of Computer Science & Engineering, ITER, Siksha 'O' Anusandhan University
Bhubaneswar, Odisha, 751030, India**

**[2] Dept. of Computer Science & Engineering, ITER, Siksha 'O' Anusandhan University
Bhubaneswar, Odisha, 751030, India**

**[3] Dept. of Computer Science & Engineering, ITER, Siksha 'O' Anusandhan University
Bhubaneswar, Odisha, 751030, India**

**[4]Dept. of Computer Science & Engineering, ITER, Siksha 'O' Anusandhan University
Bhubaneswar, Odisha, 751030, India**

**[5] Dept. of Information Technology, C.V. Raman College of Engineering
Bhubaneswar, Odisha, 751030, India**

Abstract

The semantic based sensor service discovery is proposed to enhance the discovery of sensor services. We know web is one of the best medium connecting the service providers with their clients. Wireless sensor service generates a large number of heterogeneous raw data, so it's a big challenge now-a-days to organize these raw data using various techniques so as to make the discovery and the selection easy and efficient. This paper extends the functionality of UDDI by introducing semantic description which is stored in the semantic repository at the same time the service gets registered. To provide the requested services a match maker is usually required. The match making algorithm in this paper is a generic semantic discovery algorithm which is not restricted only to the keyword based search rather is used to find the best possible services and the selection of the right service for the right user.

Keywords: *Wireless Sensor Service, Service Registry, Semantic Repository, Match Maker*

1. Introduction

Web services are modular, self-describing, self-contained applications that are accessible over the Internet. This is an emerging trend which has been identified as the technology for business process execution and application

integration. There are also increasing number of both publicly (external) available sensor services and sensor services only exposed internally within an organization. It is becoming a kind of mainstream middleware technology of interoperation and integration between heterogeneous applications and resource sharing in Internet environment. While considering all of these factors, it becomes a challenge for the external users or systems to discover and invoke the sensor derived data.

Various standards used by the Sensor web services like XML, SOAP, WSDL, UDDI helps software to discover and access the Sensor services available on the world wide web. However all this still need some kind of human intervention in selecting the most appropriate sensor service. Currently, the industry standards for web services are Web Services Description Language (WSDL) and Universal Description Discovery and Integration (UDDI) specifications. Web services are described using WSDL definitions and advertised in UDDI registries. The current discovery mechanism supported by UDDI is not powerful enough for automated discovery. The main inhibitor is the lack of semantics in the discovery process and the fact that UDDI does not use information in the service descriptions during discovery. This makes UDDI less effective, even though it provides an interface for keyword and taxonomy based searching. The key to semantic discovery of Web services is having semantics in the description itself and then using semantic match making algorithms to find the required services.

In this paper, we develop a framework for semantically sensor web service discovery where we incorporate the semantics and integrate it with UDDI registries. Our aim is the discovery of Web services on a semantic comparison between a client query and available sensor services. This architecture supports both service publishing and service discovery. The discovery contribution of this paper lies in four fold. First is the direct discovery by exact matching. If this step fails, automatically the requested query for the service is matched with the semantics. Third, we use a dictionary

based approach to capture real world knowledge if the second step is not successful and it will also function ate automatically if a failure occurs in the third step. The fourth and the final fold in our model will be an advanced search having its separate searching interface which will be used by a requester when he wants a particular service according to his non-functional (QoS) requirements.

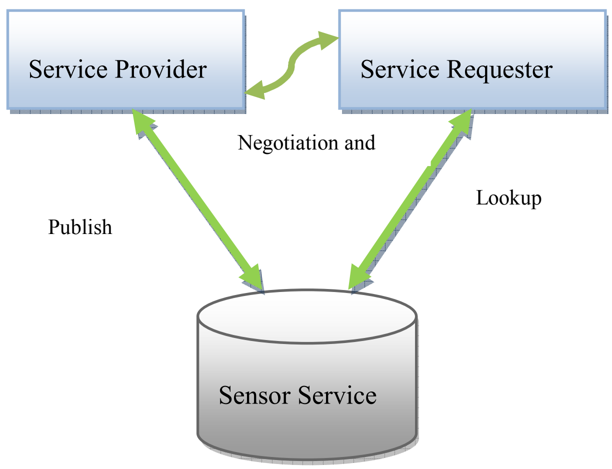

Fig. 1 Overall Architecture of Service Storage and Service Discovery.

The remaining part of the paper is organized as follows. An overview of the related work is described in Section 2. The Concepts of Semantic Repository is explained in Section 3. The architecture of sensor web registry and their attributes for sensor web service discovery is mentioned in Section 4. Section 5 presents the proposed Framework, an effective searching algorithm for sensor web service discovery based on functional and QoS requirements, the information flow between various layers of the proposed architecture, and various parameters used in our proposed model. Section 6 gives the implementation details of the proposed technique and finally section 7 presents the conclusion and future work.

2. Related Works

The authors in [1] have proposed an algorithm for an efficient search but it is limited to only keywords and also they have not implemented the algorithm.
The authors in [2] proposed several important components on the sharing of sensor network data using the Web services and SWE standards. The Sensor Web proposed a registry for sensor network discovery and registration called Sensor Registry Service. The Sensor Registry Service is too abstract in the service oriented Sensor web because too little attention has been given to the detail functionality of the sensor registry service.
In [3] a mechanism to discover sensor web registry services based on functional requirements is proposed. However nonfunctional requirements (QoS Parameters) of the services are not considered at all.
In [4] a unique SOA approach is presented to design a sensor web registry that can be hosted on a special server called Sensor Name Server that cooperates and collaborates in searching a sensor network. However the author has given more emphasis on design of sensor web registry rather than sensor discovery process. This paper explains that the client is given a web based GUI interface to search a sensor network based on single or combination of parameters given in the registry. But it may not be an efficient approach for the casual users who have no idea about the above type of sensor search interface with parametric choice.
In [5] a sensor network registry is proposed and the query parameters for sensor network discovery are analyzed by 5W1H method. Here the authors have mentioned that the sensor network registry receives the discovery query using XML (XQuery). However XQuery and XPath are the advanced XML based technology which is very difficult for the novice requesters to understand.
In GEOSS [6], a community of researchers emphasized the need of sensor web registry with SWE compliance. It also discusses ad hoc network and moving (nomadic) sensor. But It does not provide any solution for design and discovery of sensor web service.
In a similar effort, the authors in[7] proposed WOOGLE , a search engine which focus on retrieving WSDN operations. Woogle (which discontinued its service in 2006), collected services from accessible services registries and provided clients with capabilities to perform keyword-based search. However, the main underlying concept behind the method implemented in woogle was based on the assumption that web services belong to the same domain of interest and are equal in terms of their behavior in accomplishing the required functionality.
Other approaches focused on the semantic support for web services as presented in [8], the authors proposed a novel approach to integrate services considering only their availability, the functionalities they provide, and their non-functional QoS properties rather than considering the users direct request.
In [9] the authors proposed a solution for this problem and introduced the Web Service Relevancy Function(WSRF) that is used for measuring the relevancy ranking of a particular Web service based on QoS metrics and client

preferences. However one of the challenges in this work is the client's ability to control the discovery
process across accessible service registries for finding services of interest, yet semantic matching of services has not been considered.
The authors in [10] proposed an important concept of Static Discovery and Dynamic Selection(SDDS) approach to web service discovery but their approach is very tedious which will adversely affect the execution time resulting in a poor performance. Also they have not mentioned anything about the implementation of their system.

3. Semantic Repository

The traditional way of service discovery such as the UDDI discovery is keyword-based matching that is usually considered poor in performance. To enhance the efficiency and accuracy of service discovery, we propose the idea to append additional semantic information for the services in the service registry, which describe the services' details that can be used to discover appropriate services.

The main task of semantically enhanced service repository is to provide the ability of service discovery and composition based on semantic. It needs some algorithms to find out the appropriate service(s) matching client's request. If there is no such service, some more algorithms are implemented in our model in order to find out a sequence of some services that can satisfy the request, which is called service composition.

4. Sensor Service Registry

A Sensor service registry is designed with the efficient capabilities of managing services with functions such as service registration, publishing, and monitoring. The registry manages all the internal and external services.

4.1 Architecture of Sensor Service Registry

The service provider will be provided a Sensor service registration interface (Fig. 4) using which he/she can register the sensor services. When a service provider registers a new service, the registration module sends the functional and non-functional (QoS) information regarding the service to the Service Storage System (SSS). The SSS is meant to store all of the information in Sensor Service Registry except the service description which is stored separately in the Semantic Repository. When the end user sends the query message to access a Sensor service by the help of Sensor web discovery interface (Fig. 5), it is delivered to the Service Discovery System(SDS). The SDS processes the user query and retrieves the appropriate Sensor service from Sensor Service Registry along with the service description from the Semantic Repository which is finally displayed to the Service requestor.

4.2 The functions of a Sensor service registry

The service registry allows the service providers to register and update his services in the registry, which includes the followings:

1. Service Registry: The registration information includes attributes such as name, service address and other non-functional (QoS) parameters of the service.

2. Service Discovery. In the Sensor service registry, a simple search technique using the keyword is implemented, where as the service discovery discussed in the next section is the advance search techniques.

3. Service Validation. The service validation checks if the service is already available.

5. Proposed Work

Current Web service standards focus on technical Conventions. Though they solve many problems on the technical level, the semantics of Web services and Web service descriptions as a whole are not addressed by them [1]. Motivated by the increasing number of Sensor services we are here to propose a brand new approach for the service registry and discovery of these sensor services so that each and every consumers expecting some services will get the desired result always.

Our approach is very similar to that of the author's in [10],we have designed an easy and efficient algorithm along with the implementation to overcome the limitations found in [10]. We propose addition of semantic data in the present WSDL by Using a separate Semantics Repository where all the semantic information of the services will be stored. Once the service is registered and the semantics are stored, the next important issue is how to find semantic similarity between the semantic annotations and different domains. Since the UDDI approach suffers from some serious bottlenecks, our proposed work provides a series of algorithms to avoid all the difficulties faced by WSDL and UDDI.

The main focus of our framework is the intermediate layer called Service Discovery System (SDS) which provides interoperability between the service requester and Sensor Storage System. The Semantic Repository and Sensor Service Registry combined represent Sensor Storage System. Two more important operations added to the SDS of our proposed framework are Service Rate System (SRS) for finding the most suitable sensor service based on rate of service and we have also proposed an Advanced Search to find the most relevant services based on the functional and non-functional (QoS) requirements of the user. This will be the last and independent operation of the Service Discovery System.

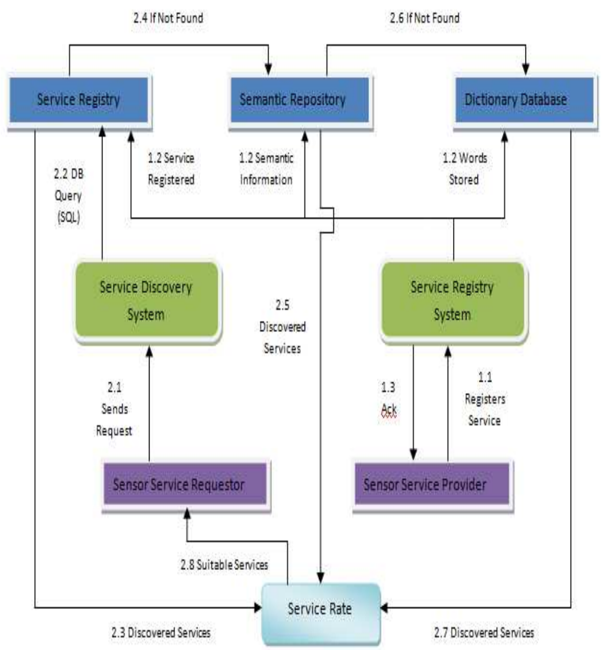

Fig. 2 Proposed Architecture for Sensor Service Discovery

In Fig. 2, the sensor service registration process is the process no 1 which starts from the sensor service provider, so the flow is represented as 1.1, 1.2 and so on. The Sensor Service discovery is the process no 2 starts with Sensor service requestor and is annotated in a similar fashion. Fig. 3 represents the sequence diagram of our proposed model.

5.1. Proposed Framework

The proposed architectural framework of our model will flow as below.

A typical usage scenario is described here by considering an example in which a Wireless Sensor service provider registers his service and a consumers request for a service.
1. Sensor Service Provider: - Initially the Sensor Service Providers will register the Wireless sensor service in the Sensor Storage System and provide functional and non-functional information (QoS) about the offered services.

2. Sensor Service Requester:- Sensor Service Requestor is the consumer who requests for the Sensor service discovery by providing his query for the service. The request made may be for Semantic Search or Advance Search.

i. Semantic search- In Semantic search, the user have to only provide his query as we usually do in Google.

ii. Advance search- In Advance Search, the user have to provide both query (functional) as well as the QoS (non-functional) parameters. This is a case usually found in job search interface of different company websites.

3. Service Discovery System (SDS):- The SDS will accept the request from the requester and scans the request query string and applies necessary algorithms along with the SQL queries to extract possible services from the Sensor Storage System.

4. Service Registry (SR):- Service Registry acts as an information registry for all the Sensor services which gets registered. All the functional and non-functional information provided by the Service provided will be stored in SR.

5. Semantic Repository (SMR):- Semantic Repository stores only the semantic information related to the services.

6. Dictionary Database (DD):- Dictionary approach is used in our model to capture real world knowledge if the user by mistakenly doesn't provide the query correctly i.e .for example if someone provides "rainf" instead of "rainfall". Our system will still display the results for rainfall without failing.

7. Service Rate System (SRS):- After the SDS applies the extraction methods onto the Sensor Storage System, a list of Sensor Web services according to the user requirements will be returned back to SRS. The SRS then arranges the services according to their Service Rate which is nothing but the frequency of a particular service. The frequency in our context is the number of times the page is accessed. So the SRS arranges the discovered services according to their frequency in a descending order.

8. Finally the organized sensor services by the Service Rate System (SRS) is returned to the user.

5.2. Proposed Algorithm

1. Request for a desired Sensor web service
2. Split words by white space and store in the array.
3. Discover Service.

i. Fire SQL query to perform a traditional keyword based search to find the requested Sensor web services. If there is no match found or if the user is not satisfied with the match results then goto Step **3.ii** else goto Step **4**.

ii. Fire SQL query integrated with codes to match those array of words with that of the semantic descriptions of services stored in semantic repository database. If found goto Step **4** else goto Step **3.iii**

iii. Fire SQL query to match the query of words with the substring of words present in the dictionary database. If found goto step **4** else displaying "No result found".

iv. After processing the above two SQL queries, a list of sensor services are discovered from the Service registry.

4. Service Rate calculation.
   The rate of service is calculated according to the service bits.
5. Invoke appropriate service
   Most relevant sensor web service invoked and is displayed according to their service rating in descending order.

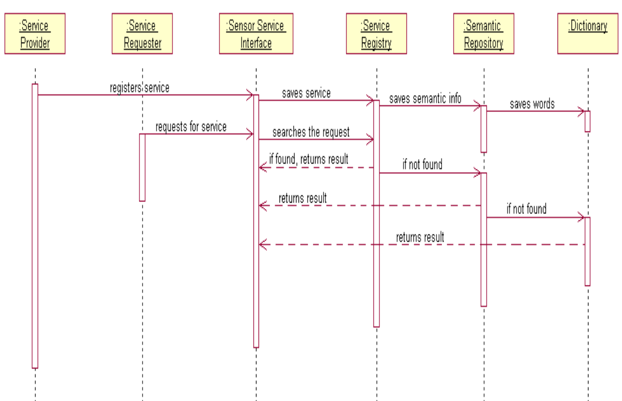

Fig. 3 Sequence Diagram of Our proposed Model

5.3 Various Parameters Used In Our Model

We should model a search engine in such a way so as to extract a satisfying result for all the requestor requesting for a service. An efficient Sensor service system should contain the functional parameters as well as the non-functional parameters for an efficient sensor service discovery.

The Functional Parameters used are the Sensor Service Name(SSN), Sensor Service Address(SSA), Sensor Service Description(SSD). The Non-functional Parameters used are QoS Data like Response Time (RT), Throughput (TP), Availability (AV), and Cost of Service(C).

6. Experimental Evaluation

The proposed system for discovering Sensor Web service is programmed using PHP technology is a distributed, loosely-coupled, Platform-independent system, which can run on multiple operating systems, such as Linux, Windows, or Solaris. The Sensor Service storage system is designed using MySQL database package.

This GUI shown in Fig.4 is the Sensor service registration interface. Using this GUI, all the Sensor service providers will register their services.

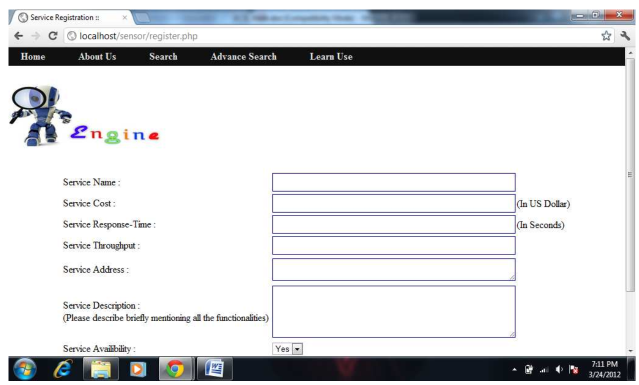

Fig. 4 Service Registration Interface for Sensor service.

Now after the Sensor service registration, to demonstrate our proposed model we need to forward some feasible test conditions for a better understanding.

Test Scenario-I: The first step of our model is to perform a Keyword based search to get the desired Sensor web service. The example is illustrated in Fig. 5.

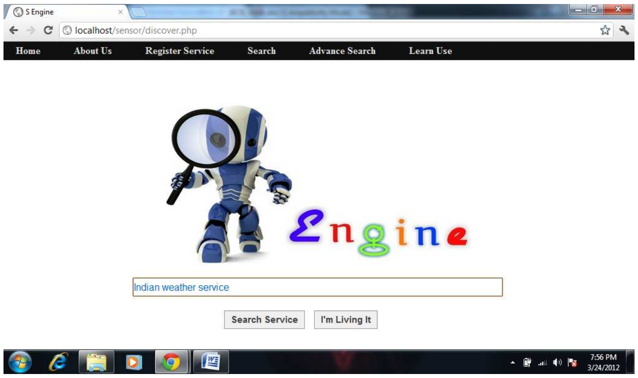

Fig. 5 Service Discovery Interface for Sensor services.

Suppose there is a Sensor service registered with name “Indian Weather Service”. So searching the Sensor service name itself will directly perform a Keyword based search to find the requested services. If found it will save the execution (processing) time incurred during Semantic Searching. So, searching in Test Scenario-I is limited to Keyword and and saves execution time.

There is a hyperlink showing “Click here to view some more related results” in Fig. 6. As it is a Keyword based search the request may not be satisfied with the extracted sensor services. So we have an option to display the Semantic results in a Single Click.

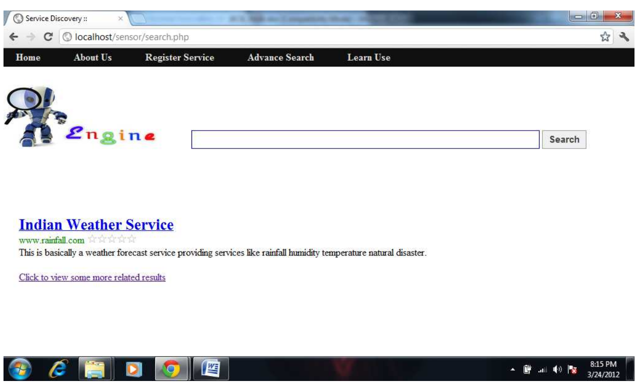

Fig. 6 Service Discovery Results for Sensor services.

Test Scenario-II: Suppose we are searching for “Weather Services” and there is no services registered with the same service name, then our model will fail in the Keyword search technique but without aborting it will move ahead to perform a semantic search using the description of all the service registered. When it will find some valid match it will display all the results according to their service rates (in a descending order). The words in the query which is matched will be highlighted as well. This Test Scenario is illustrated in Fig. 7 and Fig. 8.

Fig. 7 Service Discovery Interface for Sensor Web Services.

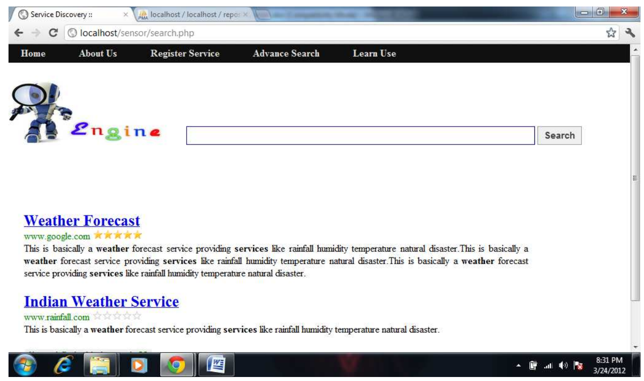

Fig. 8 Service Discovery Result for Sensor services(semantic search).

Test Scenario-III: Usually 6 out of a 10 times this case arises when a service requestor unknowingly makes a mistake while typing his query. We have used a Real World Knowledge by introducing a dictionary database for the first time which will help the user if he mistypes something and fortunately it is a substring of an entry in

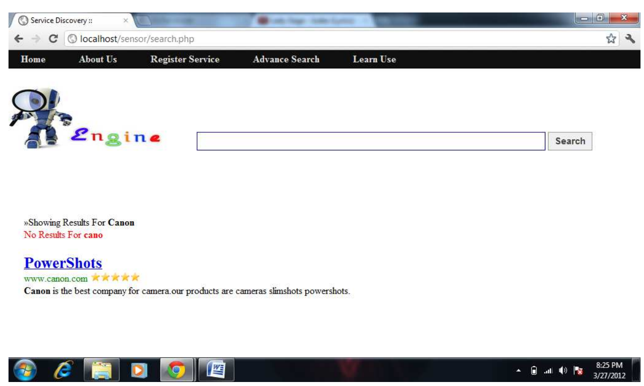

Fig. 9 Dictionary Discovery Result for Sensor services

our dictionary database. Lets consider a requestor mistypes “humid” instead of “humidity”. In this case most of the systems fail due to the lack of real world knowledge but this system will definitely return a result for sure without failing.

In Fig.9, we searched for “cano” instead of “canon” but it returned the results for “canon” i.e. why the web page says “>>Showing Results for **Canon**, No Results For *cano*”.

Test Scenario-IV: This demonstrates an Advance Search Technique basically depending on the Quality of Service (QoS) Parameters. This is the most efficient search technique to find the most relevant Sensor service not only using the QoS parameters but also it will undergo a semantic search. So this search is an improvement to that

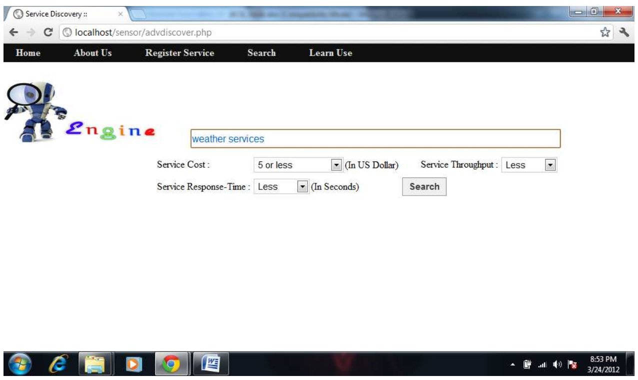

Fig. 10 Advance Service Discovery Interface for Sensor services.

explained in Test Scenario-II. This type of search is usually found in Job Search engines in company websites but they are also limited to Keyword search with QoS Parameters. The Sensor Service Requestor should provide some QoS parameters according to his need, and finally our search engine will filter out all the relevant services available in the repository. This is illustrated using Fig.10 and Fig.11.

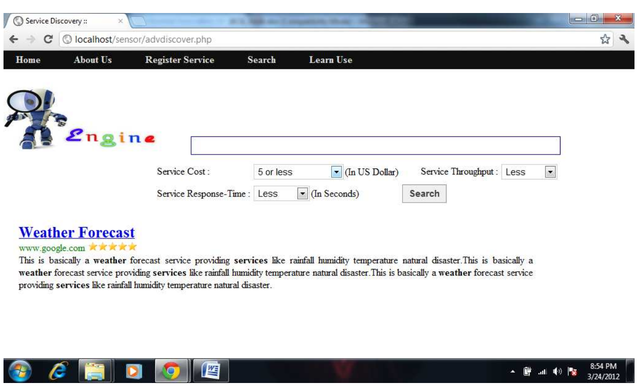

Fig. 11 Advance Service Discovery Result for Sensor services

It also undergoes a Service Rate algorithm and the Service Rate is represented by the STAR Rating near the Service Name and Service Address.

7. Conclusions

To avoid the serious bottlenecks in WSDL and UDDI registries we are successful in the completion of a system which is much more advance in dynamic service discovery and selection, added with some extra and completely new features. This paper presents a semantically enhanced Sensor Storage System consisting of two crucial parts, the Sensor service registry and Semantic repository. We have provided all possible diagrams and some important test cases for a better understanding of our model. These Test cases are most likely to occur during a Sensor service discovery. This model can be extended in multiple directions. We can integrate some security features in our model which we have not focused in our current work. An Auto- Suggester as well as an Auto-Previewer of the Sensor web service page can be added using the implementation of Ajax as an extra work in the future.

References

[1]Rajender Nath , Harish Kumar(2009), “Building Software Reuse Library with Efficient Keyword based Search Mechanism”– International Journal of Computing Science and Communication Technologies, VOL. 2, NO. 1, July 2009. (ISSN 0974-3375)

[2] Chu X., Buyya R., (2007), "Service Oriented Sensor Web, in Sensor Networks and Configuration: Fundamentals, Standards, Platforms and Applications” , N. P. Mahalik, (ed). Springer-Verlag, 2007, pp. 51-74, 978-3-40-37364-3

[3] Manoranjan Parhi, B.M. Acharya, B. Puthal “An Effective Mechanism to Discover Sensor Web Registry Services for Wireless Sensor Network under x-SOA Approach”, Proceedings of 2nd IEEE International Conference on Trendz in Information Science & Computing[TISC-2010], Chennai, India, December 17th -19th, 2010

[4] Pandey, K.K. Patel, S.V. (2009) “A Design of Sensor Web Registry for Wireless Sensor Networks with SOA Approach”, Proceedings of the 1st IEEE international conference on Computational Intelligence, Communication Systems and Networks(CICSYN2009), Indore, India, July 23-25, pp.247-252

[5] Jeongkyu Park; Jiung Han; Kibong Kang; Keung Hae Lee.,(2007), "The Registry for Sensor Network Discovery", Proceedings of the 12th IEEE International Conference on Engineering Complex Computer Systems (ICECCS2007), Auckland, New Zealand, July 11-14, pp.129-137.

[6] Ingo Simonis " Meraka Institute, CSIR, South Africa , Johannes Echterho! " IfGI, University of Muenster, Germany June 2008 , GEOSS Sensor Web Workshop Report

[7] X. Dong, A. Halevy, J.Madhavan, E. Nemes, and J. Zhang, "similarity search for web services," in proceedings of the Thirtieth international conference on Very large data bases,ser. VLDB '04 VLDB Endowment,2004,pp. 372-383.

[8] N. Ibrahim*, F. Le Mouël** and S. Frénot, "Mysim: a spontaneous service integration middleware for prevasive environments," in proceedings of the 2009 international conference on prevasive services,ser. ICPS '09,New York,NY,USA: ACM,2009,pp. 1-10.

[9] E. Al-Masri and Q.H. Mahmoud, "Discovering the best Web service," in proceedings of the 16th international conference on World Wide Web,ser. WWW '07. New York,NY, USA : ACM, 2007, pp. 1257-1258.

[10] Atousa Pahlevan and Hausi A. M¨uller, "Static-Discovery Dynamic-Selection (SDDS) Approach to Web Service Discovery," Proceedings of 7th IEEE International Conference on Web Services(ICWS 2009), pp. 769-772, July 2009.