# A Flexible tool for Discovery and Selection of Sensor Web Registry Services with Extended SOA Framework

Manoranjan Parhi Assistant Professor, Dept. of Computer Sc. & Engg. ITER, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India

## ABSTRACT

Sensor Web Services are recently emerged as promising technology and have potential usage in a wide range of application domain. However due to advancement of OGC SWE standard, the discovery and selection of Sensor Web Registry services throughout heterogeneous environment become a challenging task. Many approaches and frameworks have been proposed to discover the sensor web registry services and some of the approaches assume that the requests are placed in SOAP compatible formats while others focus on GUI based parametric query processing. In this paper, a flexible tool has been proposed that uses the Natural Language Query Processing which is a convenient and easy method of sensor data access in comparison to SQL or XML based Query Language like XQuery and XPath. An architecture based on extended SOA Framework that organizes the method of sensor web registry service discovery and selection in an efficient and structured manner has been suggested by adding some new layers like Request Parser & Query Generator (RPQ), Service Verifier and Certifier (SVC) and Service Rank Calculator (SRC). A typical weather sensor web service scenario is considered in this paper for finding the most relevant service according to requester's functional and QoS requirements.

#### **General Terms**

Sensor Web, Service Oriented Architecture, Quality of Service (QoS)

#### **Keywords**

Wireless Sensor Network (WSN), Sensor Web Enablement Standard (SWE), Sensor Web Registry, Natural Language Processing, Request Parser & Query Generator (RPQ), Service Verifier and Certifier (SVC), Service Rank Calculator (SRC)

## **1. INTRODUCTION**

Sensor Web service has emerged as a new kind of distributed calculation paradigm. It has characteristics of cross-platform, component-oriented and loose coupling, etc, and it is becoming a kind of mainstream middleware technology of interoperation and integration between heterogeneous applications and resource sharing in Internet environment. Therefore it becomes a challenge for the external users or systems to discover and invoke the sensor-derived data. It requires immediate attention to facilitate smooth search and query operations on all the sensor networks and to get a global view of the data. This will facilitate accuracy and comprehensiveness in analysis, and synthesis of the data. The SWE framework [1] which is based on SOA consists of a set of standards that define data formats for sensor data and metadata and web service interfaces for providing sensor related functionality. The SOA is the proven technologies which can be considered for design, implementation and discovery of efficient sensor web registry service. The QoS

## Biswa Mohan Acharya Assistant Professor, Dept. of Computer Applications ITER, Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India

based sensor web service discovery play an essential role in SOA because most of the applications want to use sensor services that accurately meet their requirements. This work proposes a sensor web service discovery mechanism in which the functional and non-functional requirements (QoS) are taken into account during service discovery. The general architecture of service discovery is shown in Figure 1.

In this paper an AI technique called Natural Language Query Processing [10] for human-sensor web registry interaction have been used without using any parametric choice graphical user interface. By the help of this approach the client request written in English sentence can be translated into SQL using semantic grammar for discovering corresponding sensor web service. An improvised architecture has been proposed, called extended SOA framework using an intermediary, requester friendly layer called RPQ between the service provider and service requester via a service registry that organizes the method of sensor web service discovery in an efficient and structured manner. Two new layers have been added called SVC and SRC to the proposed framework for evaluating and discovering the most appropriate sensor web service based on functional and QoS requirements.



## Figure 1: General Architecture of the Service Discovery (SOA)

The remaining part of the paper is organized as follows. An overview of the related work is described in Section 2. The architecture of sensor web registry and their attributes for sensor web service discovery is mentioned in Section 3. Section 4 presents the proposed work and covers an extended SOA Framework, an effective algorithm using NLP for sensor web registry service discovery based on functional and QoS requirements. Section 5 gives the implementation details of the proposed technique and finally section 6 presents the conclusion and future work.

## 2. RELATED WORK

Many approaches and frameworks have been proposed to design and discover the sensor web registry services. In [2] the authors 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 our earlier work [3] we have proposed a mechanism to discover sensor web registry services based on functional requirements. However non-functional requirements 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 discover of sensor registry service. The author in [7] proposed that the sensor register is part of the web framework but emphasis is given on classification of domain name system like naming for different region. It is very conceptual and does not elaborate the other contents which can be the part of the registry. The IrisNet [8] also proposes the software infrastructure for the sensor network discovery. The IrisNet is different from the SensorWeb in that it operates the independent registry for each sensor-based service. The configuration of an independent registry for the each service causes a problem because the user can not be sure of the sensor network address at the first stage of the service discovery. Trabelsi et al [9] also present sensor web registry but their primary focus is on the context based security of sensor networks. The author has mentioned about providing the security against various attacks. It is also suggested to use the Web Ontology language to classify the sensor information relationship and behaviors. The ontology is still the state of art and lot of research is going in this area, and will be really useful but again it poses a limitation on the kind of search engine and complexity of logic which one has to put in developing a registry for search. As per the above discussion a concrete proposal is yet required for the design and discovery of universal distributed sensor registry database service which can handle heterogeneous sensor networks more efficiently. The most important points of sensor network data sharing is defining the different functions of the sensor web registry such as the registered information and the request query.

#### 3. SENSOR WEB REGISTRY

In this section we describe the overall architecture and design of the sensor web registry and define the necessary querying parameters for sensor service discovery. There are two objectives of our Sensor Web Registry Services. One is to provide the capabilities to store, manage and version service information and their artifacts. The other is to define a XML Schema for service description and generates WSDL (Web Service Description Language) accordingly, when needed.

#### 3.1 Architecture of Sensor Web Registry

The schematic diagram of sensor web registry architecture and operations are explained in Figure 2. The sensor network service provider registers the services of sensor network through sensor web registration interface for sensor-derived data opening. The registration module sends the information, which is registered by the sensor network service provider to the Information Storage (IS) of sensor registry. When the user sends the query message to sensor web registry the sensor web discovery interface delivers it to the query processor. The query processor processes the user query and brings related sensor network information from the information storage of sensor registry.



Sensor Service Provider

#### Figure 2: Architecture of Sensor Web Registry

# **3.2 Query Parameter of Sensor Web Registry**

An efficient Sensor Web Registry should contain the following functional and non-functional parameters for satisfying the sensor web service discovery query efficiently. The Functional Parameters are namely Sensor Domain Name, Gateway IP Address, Port, Mobility Status, Geo Spatial Data, Meta Data, Data Type Sensed, Registration Required, Political Data, Sensor Network Description Service, Sensor Network operator. The Non-functional Parameters are QoS Data like Response Time (RT), Throughput (TP), Availability (AV), and Cost of Service(C).

## **3.3 A Sample XML document for Sensor** Network Description

The Sensor Web Service gives the overall description of the sensor network. In our service registry, a XML schema is defined for describing the services. The service registry is the master catalog for service description. The concept of service

International Journal of Computer Applications (0975 – 8887) Volume 44– No14, April 2012

here includes the traditional web services that implements WSDL interfaces with SOAP/HTTP The following XML document represents a sample for sensor description. <?xml version="1.0" encoding="UTF-8" ?> <SensorNetDescription>

<SensedEntity>

<Name>Temperature</Name>

<Unit>Centigrade</Unit>

<ProcessChainName>TemperatureService</Process ChainName>

<StarterService>http://www.wsn.env.odisha.in:85/
GetTemperature?wsdl</StarterService>
<BindingType>SOAP\_BINDING

</BindingType> <Return>**ARRAY2D-x, y, data**</Return> <PortNo>**85**</PortNo> <RegistrationRequired>**True**</RegistrationRe quired> <RolesAllowed> <Roles>**Admin,SuperVisors,Wellwishers**</R oles> <Users>**Manoranjan,BiswaMohan,**</Users> </RolesAllowed>

</SensedEntity>-

</SensorNetDescription>



Figure 3: Proposed Architecture for Sensor Service Discovery

## 4. PROPOSED WORK

The proposed architectural is inspired by extended SOA framework for the sensor web service discovery mechanism based on Natural Language Query Processing using semantic grammar, functional and QoS requirements. The main focus of our framework is the intermediate layer called RPQ which provides interoperability between the service requester and service registry. Two more layers are added to the proposed framework named SVC and SRC for certifying the sensor web service based on QoS factors before publishing in sensor web registry and finding the most suitable sensor service

based on rank of service respectively. Also an algorithm for various stages involved in the process of sensor web service discovery is proposed.

## 4.1 Extended SOA Framework

The proposed architectural framework (Figure 3) is an enhancement over the existing 3-tier SOA (Figure 1) architecture consists of

- Sensor Service Requester
- Request Parser and Query Generator

- Service Rank Calculator
- Sensor Service Registry
- Service Verifier and Certifier
- Sensor Service Provider

A typical usage scenario is described here by considering an example in which a consumer uses a weather sensor web service of a provider in its application.

**1.** Initially the weather sensor web service providers register the weather sensor web services with the Sensor service registry via Service Certifier and Verifier and provide functional and non-functional information (QoS) about the offered services. QoS parameters help determine which of the available Sensor Web services is the best and meets clients' requirements. Because of their significance, we selected the following QoS parameters that are used for computing Score values of Sensor Web Service.

**Response Time** (RT): the time taken to send a request and receive a response (unit: milliseconds).

**Throughput** (TP): the maximum requests that are handled at a given unit in time (unit: requests/min).

**Availability** (AV): a ratio of the time period when a Sensor Web service is available (unit: %).

**Cost of Service** (C): the cost per Senor Web service request or invocation (\$ per service request).

 
 Table 1: Functional & QoS Measurement data detected by the weather sensor service

Functional & QoS Measurement	Туре	Unit
Timestamp	Long	Milliseconds
Location	String	-
Date	Date	-
Weather Condition	Integer	-
Temperature	Double	Degrees Celsius
Due point	float	Degrees Celsius
Humidity	float	Percent (%)
Pressure	float	mb (Millibar)
Wind Speed	float	Km/hr
Wind Direction	float	Degree
Response Time	Integer	Ms (Milli Second)
Throughput	Integer	Req/min
Availability	float	Percent (%)
Cost	Dollar	\$/request

Table 1 shows the functional and QoS measurement data provided by weather sensor web service.

2. The Service Verifier and Certifier (SVC) verify the QoS information of the weather sensor service and issue a certificate. A copy of the QoS certificate is stored in the QoS database of Sensor Service Registry and a copy is sent to the Sensor service provider. This QoS certificate assures that the QoS offered by the provider confirm to their descriptions. The service provider initiates the verification process through the service publisher by supplying the QoS property values. The verifier is provided with the WSDL document and additional information about resources available at the provider's platform. The verifier performs the testing of the service URI, the XML schema definition, the service binding information and the availability of all operations described in the service interface. Verifier also performs the verification of the QoS information introduced in the service interface. The QoS verification is conducted through a set of test cases generated by the verifier. For each test, additional information like server capacity, network bandwidth about the provider and its web service are needed. The four QoS parameters [14] [15]

(Response Time, Availability, Throughput, and Price) are also verified. The verification process is done in three levels:

- General web services information verification
- WSDL content verification
- QoS verification.

A web service is said to be compliant with a given level when it passes the corresponding tests described in the verification document. Based on this, web services can be classified into three classes. Class A includes web services for which all verification tests have succeeded. Class B includes web services for which more than 80% of the verification tests have succeeded. Class C contains the services for which most of the verification scenarios have failed. Once the verification process is completed successfully, the certification process is initiated. The certifier issues a certificate to the service provider through the service publisher which indicates that the offered QoS confirm to their descriptions. The main responsibility of the certifier is to certify the web services and their provided QoS. A copy of the certificate sent to the service provider, which is also stored in the WSS for future use. The certificate includes information such as certificate number, certificate issue date, number of years in business and service location. In case, if the certificate cannot be issued, feedback will be sent to the provider. After the QoS certification process, the service publisher can register with the Sensor Web registry, the functional description of the service and the certified OoS information.

3. The consumer application requests the weather sensor web service discovery and provides functional and QoS requirements by the help of natural language (English Query).4. The RPQ will accept the request from the requester and parse the request query string into its technical form by

deriving the semantics of the natural language query. 5. After successful parsing of the statement given by the

requester, the RPQ generates a query against the user statement in SQL and further passes on to information storage of sensor service registry.

**6.** After processing of the SQL statement a list of weather sensor web services according to the required service functionality and QoS requirements will be returned back to Service Rank Calculator (SRC). The list of weather sensor web services with QoS values are shown in Table 2.

Table 2. QoS Parameter values of Weather Sensor services

Weather Sensor Web Service Provider	Throughput (Request/mi nute)	Response Time (mille second)	Availa bility (%)	Cost (\$)
S1	4	1200	80	2
S2	2	750	87	0
<b>S</b> 3	9	920	95	1
S4	6	350	90	3
S5	8	250	99	4

**7.** Service Rank Calculator (SRC) calculates rank of list of sensor web services retrieved from sensor web registry based on QoS requirements. To further distinguish qualified services, this study applies multiple criteria decision making (MCDM) with a weighted sum model (WSM) [11] [12] as a uniform evaluation method. This method is carried out in two steps.

(i) Normalization of attribute value: To prevent inaccurate evaluations due to the various measurement metrics of QoS attributes, attribute values must be normalized to the same scale. Numeric QoS attributes can be classified as either positive or negative. For a positive attribute, e.g. throughput, availability a higher value indicates better quality. A negative attribute, e.g. response time, cost exhibits the opposite effect. Correspondingly, attribute values can be normalized by the following equations where q.max and q.min are the maximal and minimal attribute value among all qualified services, respectively, and q.value refers to the attribute value of a service.

For positive attribute

$$q.value = \begin{cases} \frac{q.value - q.min}{q.max - q.min} & if q.max - q.min !=0 (1) \\ 0 & if q.max - q.min=0 \end{cases}$$

For negative attribute

$$q.value = \begin{cases} \frac{q.max-q.value}{q.max-q.min} & if q.max-q.min !=0 (2) \\ 0 & if q.max-q.min=0 \end{cases}$$

Table 3. Normalized QoS Parameter values of Weather Soncor corvico

	Sense	or services		
Weather Sensor Web Service Provider	Throughput (Request/mi nute)	Response Time (mille second)	Availab ility (%)	Cost (\$)
	0.005	0	0	0.7
SI	0.285	0	0	0.5
S2	0	0.473	0.368	1.0
<b>S</b> 3	1	0.294	0.789	0.75
<u>S</u> 4	0.571	0.894	0.526	0.25
<b>S</b> 5	0.857	1.0	1.0	0

(ii) Weighting and sum of attribute value: Based on the normalization above, the QoS performance of each qualified service can be calculated uniformly by summing up the product of each normalized attribute value and its corresponding weight as shown below. (3)

 $Score(S) = \sum_{i=1}^{n} (q_i \cdot value * w_i)$ 

As each attribute value is normalized by Eqs. (1) and (2), the sensor web service with the largest evaluation score has a higher quality than others. In the presence of a specific threshold, a sensor web service is selected as soon as its evaluation score reaches the required level. If no sensor web service reaches that threshold, there is no service available in the sensor web registry that can satisfy the service requester's requirements.

Test Scenario-I: Service Rank calculation based on Throughput and Cost (Weight of Throughput (60%) and Cost (40%)) . . . . . . .

Table 4. Service	Kank Calculation	1 for Test Scenario-I
Weather Sensor Web Service	Score Value	Rank of Sensor Web Service
Provider		
S1	0.3714	5
S2	0.4000	4
S3	0.8999	1
<u>S</u> 4	0.4428	3
S5	0.5142	2

Test Scenario-II: Service Rank calculation based on Response Time and Cost .Weight of Response Time (60%) and Cost (40%))

#### Table 5. Service Rank Calculation for Test Scenario-II

Weather Sensor Web Service Provider	Score Value	Rank of Sensor Web Service
S1	0.2000	5
S2	0.6842	1
<b>S</b> 3	0.4768	4
<b>S</b> 4	0.6368	2
S5	0.6000	3

## 4.2 Proposed Algorithm

The algorithm describes various phases for a complete cycle of a sensor web registry service discovery which is based on the proposed framework. The overall algorithm can be broken down into seven phases.

**Raise Request** i.

- Parse Request and Generate SQL ii.
- iii. Discover service
- Service Rank Calculation iv.
- Service Verification and Certification V.
- vi. Service publish
- vii. Invoke appropriate service

#### Algorithm: Sensor Web Registry Service Discovery Mechanism (SWRSD)

Input: Natural Language Request Statement Output: Appropriate Sensor Service

i. Raise Request

Let S be the natural language statement raised as a request  $S = \{W\}$  set of words

- ii. Parse Request and Generate SQL
  - 1. Input the Natural Language Query
  - 2. Parse the NL Query
    - 2.1 Split the NL by "where" into Query 1 and Query 2
  - 2.2 Split Query 1 into words by a white space
  - 2.3 Split word[i] by a comma
  - 3. Replace operator by operator symbol
  - 4. If word[1] is all then

query=select +"\* from table name where"+ Query 2 + ";";

else

query=select +" "+word [1] +"from table name where "+ Query2 +";";

- 5. return query
- iii. Discover service
  - The Sensor Service Registry has information storage.
  - The SQL Query generated in the previous step is passed to the information storage.
  - After processed by the Query processor a list of sensor services based on functional and QoS requirements are discovered from Sensor Registry.
  - iv. Service rank calculation
    - The rank of Sensor Web Service is calculated based on multiple criteria decision making (MCDM) with a weighted sum model (WSM) by applying equation (1), (2) and (3).
  - v. Service Verifier and Certifier
    - Service Verifier and Certifier perform the verification of the QoS information supplied by the Sensor service provider and issues a Certificate to the service provider.
  - vi. Publish Service
    - The verified and certified sensor network service is registered/ published in Sensor service registry if it is not available

- vii. Invoke appropriate Service
  - Most relevant sensor web service based on ranked calculation is invoked by the requester.

#### **5. EXPERIMENTAL EVALUATION**

The Natural Language Interface system for discovering sensor registry service is programmed on JSP technology [13] is a distributed, loosely-coupled system, which can run on multiple operating systems, such as Linux, Windows, or Solaris. The Information Storage of Sensor Web Registry is designed using MySQL database package. By the help of the above system a snapshot of the query is tested in the following way.

**Natural Language Query:** list sensorserviceprovider,wsdladdressofsensorobservationservice, throughput,responsetime,cost,availability where sensortype equals 'weather' and responsetime less\_than\_equals 1500 and cost less\_than\_equals 5.

GeneratedSQLQuery:selectsensorserviceprovider,wsdladdressofsensorobservationservice,throughput,responsetime,cost,availabilityfromsensornetworkregistrywheresensortype'weather'responsetime <= 1500</td>andcost <= 5;</td>

#### Final Result obtained by the User:

A list of sensor web services are obtained by the execution of above query. After rank calculation the most relevant sensor web service is obtained .

To demonstrate the effectiveness of ranking technique, consider different combination of QoS requirements (Test Scenario-I and Test Scenario-II) of weather sensor web services. The complete implementations of both scenarios are shown from Figure 4 to Figure 9.



Figure 4: Natural Language Interface for Sensor Web Service Discovery

it Vaited 🥮 Getting Started 🔝 Latest Hea	dines				
Sensor Registry Service Dicovery	*				
	-				
nerated SQL Query: se	lect				
sorserviceprovider.wsc	Iladdressofsensorobservationservice.th	roughput.re	sponsetime.co	st.ava	ulabilit
nonser treepro tider, the	nudulesse is ensered a set enter, in	a congriparite	oponioetime,ee		macin
nsornetworkregistry wh	ere sensortype = 'weather' and respons	etime <= 15	500 and $cost <$	= 5;	
sensorserviceprovider	ws dladd ress of sensor observations ervice	throughput	responsetime	$\operatorname{cost}$	avail
sensorserviceprovider s1	wsdladdressofsensorobservationservice http://www.wsn.weather.india:80/GetWeather?wsdl	throughput 4	responsetime	cost 2	avail: 80
sensorserviceprovider <sup>51</sup> <sup>52</sup>	wsdladdressofsensorobservationservice http://www.wsn.weather.india.80/GetWeather?wsdl http://www.wsn.env.india.85/GetWeather?wsdl	throughput 4 2	responsetime 1200 750	cost 2 0	avail: 80 87
sensorserviceprovider =1 =2 =3	wsdladdressofsensorobservationservice http://www.ws.weather.india/80/GetWeather?wsdl http://www.ws.env.india/85/GetWeather?wsdl http://www.ws.wether1.india/80/GetWeather?wsdl	throughput 4 2 9	responsetime 1200 750 920	2 0 1	avail: 80 87 95
sensorserviceprovider s1 s2 s3 s4	Wsdladdressofsensorobservationservice http://www.wm.weather.india.80/det/Weather?wsdl http://www.wm.wether1.india.80/det/Weather?wsdl http://www.wm.wether2.india.80/det/Weather?wsdl	throughput 4 2 9 6	responsetime 1200 750 920 350	cost           2           0           1           3	availa 80 87 95 90
sensorserviceprovider s1 s2 s3 s4 s5	wsdladdressofsensorobservationservice http://www.wsn.weather.india.00/detWeather?wsdl http://www.wsn.wetherl.india.00/detWeather?wsdl http://www.wsn.wetherl.india.00/detWeather?wsdl http://www.wsn.wetherl.india.00/detWeather?wsdl	throughput 4 2 9 6 8	responsetime 1200 750 920 350 250	cost 2 0 1 3 4	avail: 80 87 95 90 99
sensorserviceprovider =1 =2 =3 =4 =5	tvsdladdressofsensorobservationservice http://www.wns.weather.india.00/OetWeather?wsdl http://www.wns.ero.india.05/OetWeather?wsdl http://www.wns.wether2.ndia.00/OetWeather?wsdl http://www.wns.wether2.india.00/OetWeather?wsdl http://www.wns.wether3.india.00/OetWeather?wsdl	throughput 4 2 9 6 8	responsetime 1200 750 920 350 250	cost         2           0         1           3         4	avail: 80 87 95 90 99

#### Figure 5: List of Sensor Web Services based on Functional & QoS Parameters



#### Figure 6: Normalized QoS values of Sensor Web Services for Test Scenario-I



#### Figure: 7: Rank of Sensor Web Services for Test Scenario-I



Figure 8: Normalized QoS values of Sensor Web Services for Test Scenario-II



#### Figure 9: Rank of Sensor Web Services for Test Scenario-II



**Figure 10: Comparison of Test Scenarios** 

Results shown in Figure 10 can be compared to those in Test Scenario 1:- QoS Ranking Dependent on Throughput (60%) and Cost (40%) and

Test Scenario 2:- QoS Ranking Dependent on Response Time (60%) and Cost (40%).

The proposed system implementation is tested for variety of NL statements under various categories and the results obtained are satisfactory under the known constraints.

## 6. CONCLUSION & FUTURE WORK

In this paper, a flexible tool based on extended SOA Framework is presented to discover a sensor web registry services in a more structured and effective way which has many advantages in comparison with previous approaches. This tool will help to reduce the burden of novice requesters or casual users from placing the request in XML or SOAP compatible formats. Exploring the natural language semantics instead of GUI based parametric search technique make the selection of sensor web service more appropriate. Introducing RPQ, SVC, and SRC layers in our architecture offloads the message handling and preprocessing functionalities of the Sensor Service Registry. The proposed solution has shown usefulness and effectiveness of incorporating QoS parameters as part of the search criteria and in distinguishing sensor web services from one another during the discovery process. It hides the system's complexity from the clients. It provides a transparent sensor service selection from the client's point of view. It assures a level of security, since the clients do not have direct access to the sensor Web Services. Finally we have given the implementation details of our proposed technique and experimental evaluation by considering a real world scenario i.e. weather sensor web service. We have compared various test scenarios in order to obtain the most relevant weather sensor web service according to user's functional and QoS requirements. However, we are still in the earlier stage of having the entire extended SOA Framework to be implemented. The future work aims at to make natural language query processing interface more dynamic so that requester can obtain the detailed and complete picture of sensor-derived data. We have planned to enhance the capabilities of the proposed architecture to handle other QoS attributes and adapting the architecture to support various types of sensor Web services.

#### 7. REFERENCES

 Mike Botts, George Percivall, Carl Reed, John Davidson., (2006), "OGC Sensor Web Enablement: Overview and High Level Architecture", Open Geospatial Consortium, OGC White Paper, OGC 06-050r2 v2, 19th July 2006

- [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 2<sup>nd</sup> IEEE International Conference on Trendz in Information Science & Computing[TISC-2010], Chennai, India, December 17<sup>th</sup> -19<sup>th</sup>, 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 1<sup>st</sup> 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] Ian Rhead, Madjid Merabti, Hala Mokhtar, Paul Fergus, Worldwide Sensor Web Framework Overview The 9th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, 23rd - 24th June, Liverpool, UK, 2008.
- [8] P. B. Gibbons, B. Karp, Y. Ke, S. Nath and S. Seshan, "IrisNet: An Architecture for a World-WideSensor Web," IEEE Pervasive Computing, vol 2, no4, Oct.-Dec. 2003.
- [9] Trabelsi, S.; Gomez, L.; Oudier, Y. Context-Aware

Security Policy for the Service Discovery Advanced

Information Networking and Applications Workshops,2007, AINAW apos;07. 21st nternational Conference on Volume 1, Issue , 21-23 May 2007 Page(s):477 –482

- [10] Androutsopoulos, I., Richie, G.D., Thanisch, P. "Natural Language Interface to Database – An Introduction." Journal of Natural Language Engineering, Cambridge University Press. 1(1), 29-81, 1995.
- [11] Angus F.M. Huang, Ci-Wei Lan, Stephen J.H. Yang "An optimal QoS-based Web service selection scheme" Journal of Information Sciences 179 (2009) 3309–3322
- [12] C.L. Hwang, K. Yoon, Multiple Attribute Decision Making: Methods and Applications, Springer-Verlag, 1981.
- [13] Watson, Mark "Practical Artificial Intelligence Programming in Java", 2002

International Journal of Computer Applications (0975 – 8887) Volume 44– No14, April 2012

- [14] T.Rajendran, and P.Balasubramanie, (2009) 'Analysis on the Study of QoS-Aware Web Services Discovery', Journal of Computing, Vol. 1(1), pp. 119-130.
- [15] Angus F.M. Huang, Ci-Wei Lan, Stephen J.H. Yang (2009) 'An optimal QoS-based Web service selection scheme' Journal of Information Sciences pp. 3309–3322.