Mansoura Engineering Journal, (MEJ), Vol. 36, No. 4, December 2011. E. 1

A Future Image For Web Services' Discovery with A Client Web Based Interface
صورة مستقبلية لإستكشاف خدمات الويب مع واجهة رسومية للمستخدم

Engineer/Amal Yousief , Dr/Hesham Arafat , Dr/Ahmed Saleh Department of Systems and Computers Engineering , Faculty of Engineering , Mansoura University, Mansoura, Egypt
ayma_64@hotmail.com , h_arafat_ali@mans.edu.eg , aisaleh@yahoo.com

الخلاصة:
إن عملية إستكشاف خدمات الويب من المواضيع الهامة المتصلة بخدمات الويب بداية من بناء الجمل و عملية تشابه الكلمات إلى الويب الإستدلالية و القادرة على وصف الأشياء بطريقة يمكن للحاسب الآلي فهمها مع الأخذ في الإعتبار معايير كفاءة الخدمة للإختيار بين الخدمات و التي تتشابه في الوظيفة للوصول إلى أفضل خدمة ويب و التي تحقق متطلبات المستخدم .في هذه المقالة تم وضع صورة مستقبلية لما يتوقع أن يكون عليه عملية إكتشاف خدمة الويب حيث سيكون هناك دمج بين محركات البحث و مسجلات الوصف و الإستكشاف العالمية حيث أن الإتجاه هو بناء وحدة تخزين مركزية لتخزين جميع خدمات الويب بعد عملية تجميعها من مختلف السجلات و محركات البحث المختلفه. تم تغطية وجهين من الصورة المستقبلية التي تم وضعها و هما وجه4،وجه6 في الشكل رقم(3)الأول يقوم بعملية تصنيف لخدمات الويب بعد تجميعها و قبل تخزينها و يكون التصنيف إما عن طريق التشابه في الوظيفة أو عناوين الإنترنت التي تحويها يمكن ايضا الإستفادة من عملية التصنيف في بناء دلائل ويب مفتوحة تشبه تلك التي تم بناؤها لتسهيل عملية البحث عن مواقع الإنترنت .أما الوجه6 فهو عبارة عن واجهة رسومية للمستخدم يمكن إستخدامها للوصول إلى خدمات الويب المخزنة داخل قواعد البيانات و إستكشافها بسهوله حيث سيتمكن المستخدم عن طريق هذه الواجهة الوصول إلى خدمة الويب الأفضل والتي تلبي متطلباته بناء على معايير الكفاءة(QoS). تم إستخدام مجموعتي بيانات مختلفتين الأولى تحوي بيانات 365 خدمة ويب و المجموعة الأخرى تحوي ما يزيد عن 2500 خدمة ويب للوصول إلى النتائج التي تم عرضها داخل المقالة.

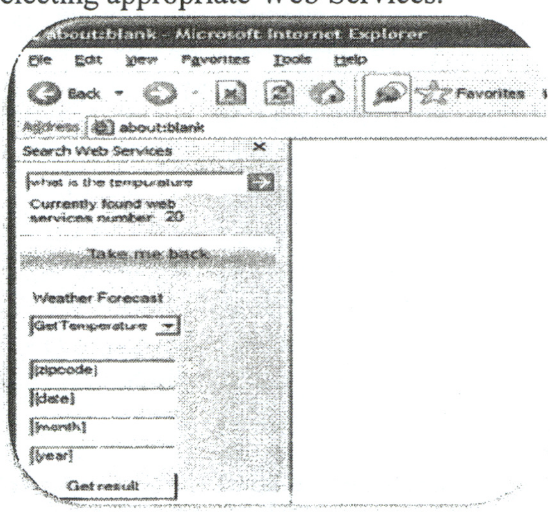
Abstract: *Web Services' discovery is a very important issue related to Web Services. From Syntax and using match-making words to semantic web and taking QoS' parameters into account for selecting between Web Services having the same functionality for finding the best service that fulfills the customer's requirements . In this paper we put a future image for Web Services' Discovery by merging both UDDI and Search Engines, as the new trend in Web Services' Discovery is building a central repository storing all Web Services after collecting them from UDDI , different UBRs ... ,this central point will be a reference to the client for searching the required Web Service .Two datasets one contains 365 WS and the other 2500 WS used in the experimental work .Our work will cover two phases from the suggested model phase4 and phase6 fig.(3).Classifying Web Services before storing them will enhance the search process and it could be a step for building open web directory contain all Web Services like that used for searching web sites for a specified issue(ODP,DMOZ..), Online databases maintain a collection of structured domain-specific documents dynamically generated in response to users' queries instead of being accessed by static URLs. We also proposed a client GUI that will will enable the Web Service consumer to easily access data stored inside these databases that contain updated frequently data of Web Services information collected ,classified and stored using different crawlers. This will facilitate and enhance the Web Services' Discovery process ,client will be able to select between Web Services due to QoS requirements and find the best Web Service that fulfills his/her requirements.*

Accepted October 12, 2011.

Keywords: *Web Services Discovery , Semantics Web Services , WSCE,GUI, UDDI, WSDL , Search Engines , Crawlers , QoS, Classification.*

1. Introduction

The service discovery process can be divided into two phases: the first one is based on the functional aspects of the service (i.e. input and output parameters, what the service does, preconditions) while the second one deals with non-functional parameters (i.e. QoS). Web Service, the magic word which will give a bright future for communications through the internet, Web Services are hardware, programming language, and operating system independent, it will move us from hardware distributed systems to software distributed systems.Talking about Web Services' Discovery problem given a repository of Web services , and a query, automatically finding a service from the repository that matches these requirements is the Web Services' Discovery problem. This problem comes in two flavors: Syntactic and Semantic[9], depending on the type of service descriptions provided in the repository, In a Syntactic Discovery WSDL provides syntactic description of Web Services which can be provided in a repository. Given a query with requirements of the requested service, the discovery problem involves finding a specific service that can fulfill the given input and output criteria in the query based on a syntactical equivalence of the input and output names. In Semantic Discovery We assume that a directory of services has already been compiled, and that this directory includes a USDL description document for each service. A USDL description of the desired service can be written, a query processor can then search the service directory for a "matching" service Quality of Service, or QoS, is "a combination of several qualities or properties of a service" and helps us to select a proper Web-service from the web applications. It is a set of non-functional attributes that may influence the quality of the service provided by a Web service like Availability,Capacity,Reliability,Performance, Cost,Response Time. There are two major problems in dynamic Web Services' discovery with QoS. The first involves the specification of QoS information. How should the QoS information be expressed and/or stored? A standard format must be agreed upon and used in order for the information to be exchanged and interpreted. The second problem is one of matching the customer's requirements with that of the provider. For example, if a customer is looking for services that matches its QoS requirements of 2ms response time, 400Kbps throughput and 99.9% availability, how can services be found whose QoS advertisement satisfies these requirements? [13].The interest of is paper is the second problem . Collecting Web Services data is not the key element that leads to an effective Web Services' discovery, but how it is stored. The fact that Web Services data is spread all over existing search engines databases, accessible UBRs, or file sharing platforms does not mean that clients are able to find these Web Services without difficulties. However, making this Web services data available from a standard, universal access point that is capable of aggregating this data from various sources and providing clients to execute search queries tailored to their requirements via a search engine facilitated by a Web Service crawler engine or WSCE is a key element to enhancing Web Services' discovery and accelerating the adoption of Web services[25] . No one before manipulated constructing a GUI which will help client side for Web Services' Discovery except at[33] fig.(2) which is a A programmable Explorer Bar for Microsoft's Internet Explorer Web Browser. The GUI will support the client side , Searching for Web Services based on QoS parameters, schema properties, service reputation, trust, and semantic matching will considerably increase the relevancy of finding and selecting appropriate Web Services.



Figure(2) A programmable Explorer Bar for Microsoft's Internet Explorer Web Browser[33]

2. RELATED WORK

Many studies , approaches and ontologies manipulated Web Services' Discovery with respect to QoS. The most of the proposed frameworks and architectures use UDDI registry for storing Web Service descriptions. When Web Services were in hundreds all approaches for Web Services' discovery were concerned with UDDI and URBs Registries when number of Web Services increased and become in thousands a new trend for Web Services' discovery depending on Search Engines, for solving the problems with Web Services' discovery and using UDDI a tModel[14] was suggested for storing values of QoS for each Web Service. A reputation-enhanced [16] model where service matching, ranking and selection algorithm is presented and evaluated . In spite of newer technologies, service registries still provide the foundation for cataloging and classifying Web Services. The UDDI Business Registry (UBR) is the central service directory for publishing technical information about Web Services, but the existing UDDI specification has some major technical limitations that make it an incomplete solution for Web Service discovery[17].Due to *UDDI* Limitations which are(i)UDDI wasn't intended to serve as a search engine for Web Service discovery.(ii) UDDI registration is voluntary and thus can easily become passive.(iii)UDDI doesn't provide any guarantee of the validity or quality of information it contains.(iv)A disconnection exists between UDDI and the current Web. UDDI is incapable of providing quality-of service (QoS) measurements for registered Web Services.(v)UDDI doesn't maintain or provide any Web Service life-cycle management. So, Search Engines might be a good alternative to using UBRs for Web Service discovery, particularly when considering information accuracy. Search Engines are trends for finding and discovering Web Services also emerged in recent years. Search Engines such as Google, Yahoo, AlltheWeb and Baidu have become a new source for finding Web services.[20] this new trend is due to limitations found with services registries like UDDI[23]. Search Engines have become a new major source for searching for Web Services. Yet, they are vulnerable to returning irrelevant results and only provide access points to WSDL documents while UDDI business registries provide a more business-centric model that can be used as the first step towards an application-centric Web , so merging between both UDDI and Search Engines in the process of Web Services discovery is the future trend. The UDDI registry can be supported with external database , which stores non-functional information about Web Services. Ran S. proposed a model by introducing a Web Service QoS Certifier module[26].Gang YE, Chanle WU, Jun YUE, Shi CHENG introduced A QoS-aware Model for Web Services' discovery by suggested a new UDDI registry which is a repository of registered Web Services with lookup facilities[27] ,the proposed new registry differs from the current UDDI model by having information about the functional description of the Web Service as well as its associated QoS registered in the repository, but their model does not modify the standard UDDI interface and the client side. Haihua Li, Xiaoyong Du, Xuan Tian[28] constructed a SAM-based service model for Web Services management, SAM-WS, and proposed an approach for users to explicitly describe their QoS requirements using QoS ontology. Eyhab Al-Masri and Qusay H. Mahmoud [29] proposed a novel exploration engine, the Web Service Crawler Engine (WSCE). WSCE is capable of crawling multiple UBRs, and enables for the establishment of a centralized Web Services' repository which can be used for large-scale discovery of Web Services they didn't cover the client side too. The proposed by Julian Day system [30] consist of two parts: augmented client and the QoS forums. Clients send their experiences to a central Web Service which stores this information inside an internal database. This Web Service can be thought of as a kind of forum system for QoS information. It can respond to requests about particular Web Services, sending all the data it knows about a particular service to a requesting client. Now when a client wants to pick a service, he/she gathers information from the QoS forums, and then reasons about which service is best. Authors at[31] suggested an ontology named DAML-QoS; it is a complementary ontology that provides detailed QoS information、for DAML-S (Darpa Agent Markup Language for Services) users. A programmable Explorer Bar for Microsoft's Internet Explorer Web Browser[33] that uses Common Sense Reasoning to display contextually relevant tasks based on what the user is viewing , and allow users to find and directly query Web Services. The Web Services Explorer Bar contains two areas Search Web Services and Tasks. The Search Web Services area allows users to query SOAP based Web Services using natural language. The Tasks area displays contextually relevant tasks based on what Web page the user is viewing.

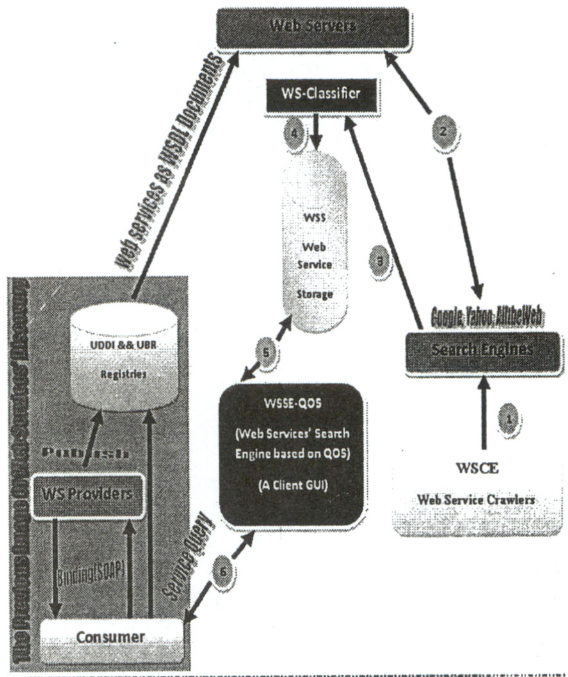
3. Aproposesd Enhanced Technique For Web Services' Discovery.

This work could be summerized into three steps (1)Classifying Web Services due to URL or a function. (2)Storing Web Services information both functional and non functional(QoS).(3)building a client web based (GUI)for Web Services' discovery.

Fig.(3) represents a future image of Web Services' discovery process based on QoS by merging UDDI and Search Engines for enhancing the discovery process through six phases our research is concerned with phase four and six , phase4 is a WS-Classifier which classified the collected Web Services in phases 1 and 2 and 3(crawlers) before storing them in the Web Services Storage, phase 6 is a Web Service Storage Search Engine based on QoS which will be a GUI to the client for getting the best Web Service which he/she searches for.

1.Classifying Web Services due to URL or a function: Classifying Web Services after fetching their WSDL documents and storing them will enhance Web Service discovery as Web Services with the same function grouped together after that the best Web Service that fulfills consumer requirements will be selected with respect to QoS (Quality of Services Parameters) according to URL or function. A new technique for managing the storage process of WSS.After classifying Web Services was implemented, the structure of the database that store data of Web Services like Name, Function, WSDL and QoS values must be changed to a new one for matching the decision tree classifier (Decision trees are powerful and popular tools for classification and prediction). Decision trees are powerful and popular tools for classification and prediction. The attractiveness of decision trees is due to the fact that, in contrast to neural networks, decision trees represent rules. Rules can readily be expressed so that humans can understand them or even directly used in a database access language like SQL so that records falling into a particular category may be retrieved fig.(4).

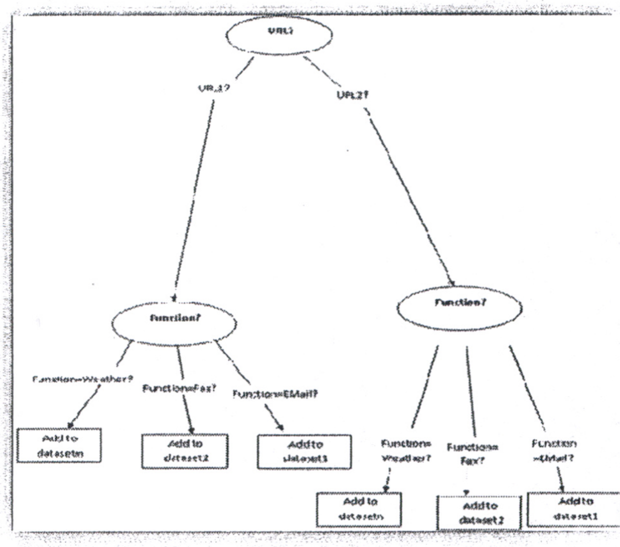
(2)Storing Web Services information both functional and non functional(QoS):
Storing Web Services information both functional and non functional(QoS) was the second step after the classyfication process by transfeering the two data sets into data bases using My Sql making the fetch process more easier,for both the provider and the client.



Figure(3)An Enhanced Model For Web Services' Discovery By Merging UDDI And Search Engine

The new technique for the storing processs will help in (i)Crawlers' function Support , using the new technique for WSS Management will support the function of crawlers , storing web services' information and metadata will be easier also help updating Web Services' information like QoS (ii)WSSSE-QoS' function Support ,WSSSE-QoS (Web Service Storage Search

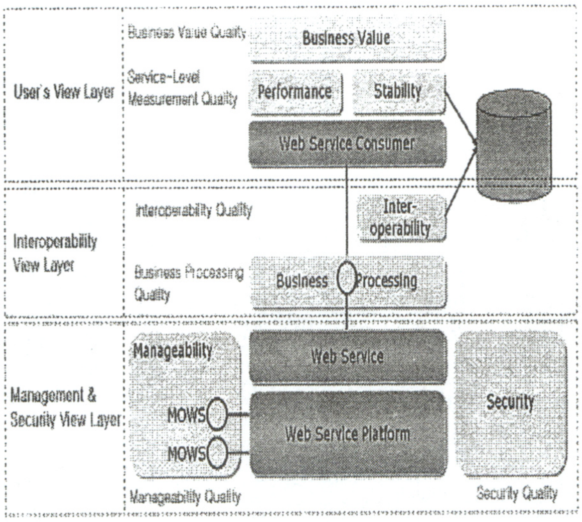
Engine based on QoS) is a system that will be established as an intermediate between WSS and consumer used for selecting the best web service that fulfills the consumer requirements due to QoS, the new technique for WSS Management will support the operation of WSSSE-QoS, which will facilitate the Find and Select operations and make them easier which will enhance web service discovery.(iii)WSS Stability ,the new technique of WSS Management will affect stability of WSS because the new infrastructure could be implemented using data structure (using trees) as storing data inside WSS through the indexing module IM , which is primarily responsible for building data structures over textual information contained within WSDL interfaces or UDDI objects(i.e. business Entity, business Service,binding Template, tModels, among others)[22].(v)Ability of Large Extension The new infrastructure of WSS will be suitable for large Extension, because tree structure has this advantage and this will be suitable for the future growth of web services numbers in the future.For implementing the new technique discussed above web directories are more suitable for storing web services data.Decision Trees was the most suitable algorithm for the suggested technique, which is a classifier in the form of a tree structure .



Figure(4) A decision tree for WS-Classifier

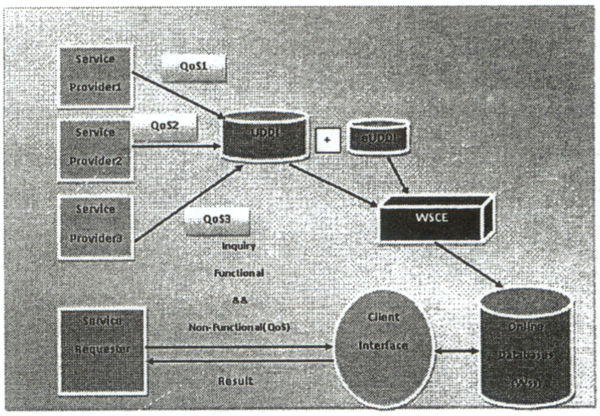
(3)building a client web based (GUI)for Web Services' discovery:

Taking into account phase 6 which is a GUI based on QoS supported the client side QoS for Web Service applications is the ability of their services to provide added value to the best solution for requesters' enquiries, with a specific requirements. QoS parameters help determining which of the available Web Services meets clients' requirements. In a previous work we used two datasets one of them consists of 365 web services, and the other one contains more than 2000 web services ,two databases were built using data of these two datasets on MySQL Server.



Figure(5)Web Service Quality Factors

Taking into account User's View Layer in fig.(5), the service discovery process adopts keyword-matching technology to locate published Web Services, basing this matchmaking on syntax level, that is, matching according to a set of weighted keywords, a less-than-desirable situation. The returned discovery result might not satisfy the requester's intended requirements. This leads to a bit of manual work to choose the proper service according to its semantics. From the Web Services point of view, the selection criteria should at least include the service's functional and non-functional requirements. To fully integrate service discovery, these domain-specific criteria should be clear and processed automatically. This requires domain-specific knowledge. QoS of any Web Service and for those which included into the datasets used[25] . fig(6) represents the discovery process where different service providers provide different Web Services with a same function but with different QoS attributes ,the functional and non-functional information of these Web Services are collected from different UBRs ,UDDI and web sites by crawlers that could be stored inside online databases, using the client interface service requester could easily get the best Web Service for him/her.



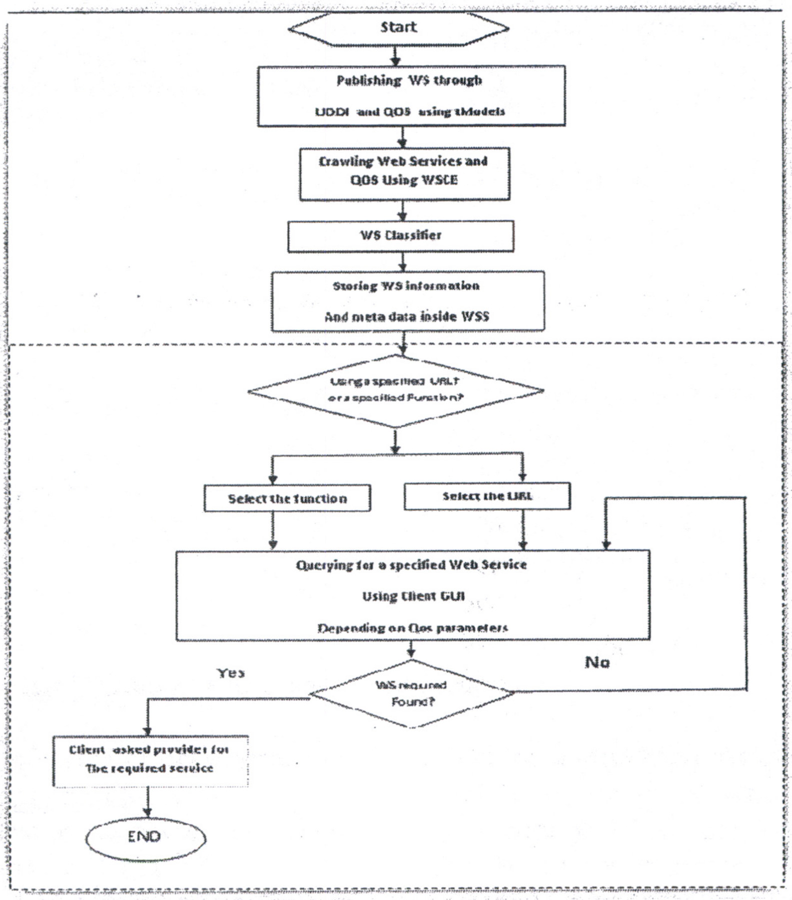
Figure(6) The discovery process based on QoS

Description of QWS dataset : QWS dataset consists of different rows of web service implementations and their attributes as presented below (http:// www.uogue/ph.ca/~qmahmoud/qws/index.html).

The attributes used in our dataset are as follows:

1. Response Time: time taken to send a request and receive a response (ms)
2. Availability: number of successful invocations/total invocations (%)
3. Throughput: total number of invocations for a given period of time (#/sec)
4. Successability: number of response / number of request messages (%)
5. Reliability: ratio of the number of error messages to total messages (%)
6. Compliance: extent a WSDL document follows WSDL spec. (%)
7. Best Practices: extent a Web service follows WS-I Basic Profile (%)
8. Latency: time taken for the server to process a given request (ms)

9.Documentation:measure of documentation (i.e. description tags) in WSDL (%)
10.WSRF:WebService Relevancy Function: a rank for Web Service Quality (%).
11.Class:levels representing service offering qualities(1 through 4).
12. Name: service name.
13. WSDL: WSDL file location.
Fig(7) is a flow chart represents the discovery process. Client could select between all Web Services due to a specified URL or a specified function then he/she could query from the returned result using QoS parameters to get Web Service that fulfills his/her requirements. The dashed area fig(7)represents the discovery process ,client could select between all Web Services due to a specified URL or a specified function then he could query from the returned result using QoS parameters to get the best result



Figure(7)A flow chart for Web Services' Discovery

4. EXPERIMENTAL WORK

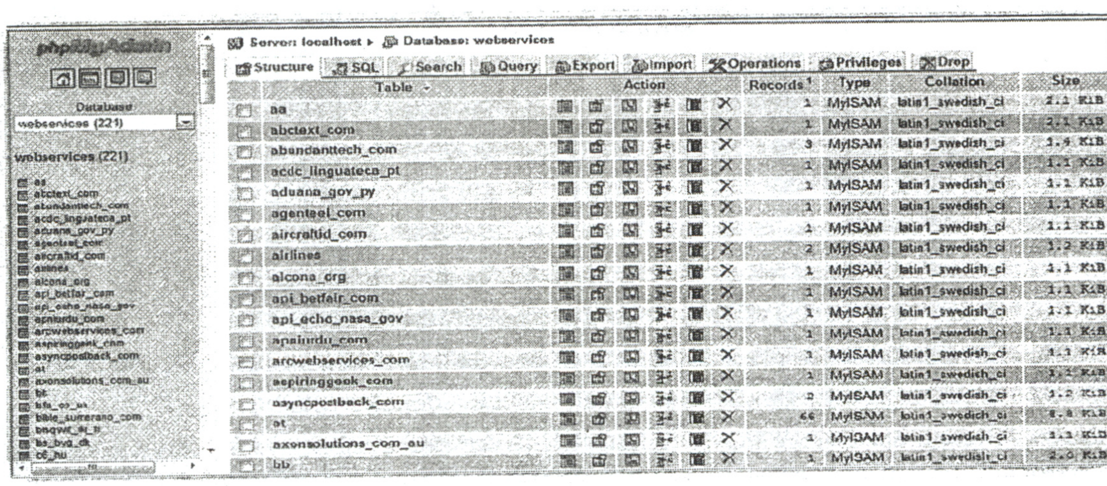
The experimental work was done in two steps (i) Implementing Web Service's Classifier System and testing the results (ii)Building a Graphical User Interface (GUI). table (1) sumarizes the experimental software used .

Connecting WAMP Server and Netbeans environment through MySQL driver as a local host and classifying data imported from a dataset contains 365 web services using SQL statements and phpmyAdmin.The output from the classifier system was 221 categories when classifying data due to URL and similar functions fig.(8).

4.1 WS-Classifier System

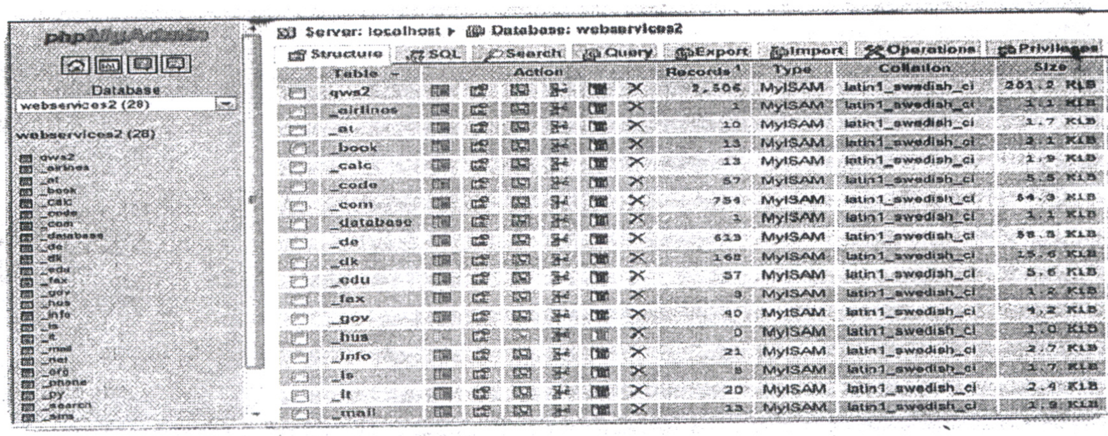
Local Server	*WAMP*
Programming Language	*Java Language*
DataBase Management System	*My SQL*
Programming Environment	*Net Beans(version 6.9.1)*
GUI design	*PHP Runner 5.2*

Table (1) Software Requirement



Figure(8)Transfering Dataset1 into a database after classification process

Using the other dataset containing more than 2000 Web Services , the output from the classifier system was 28 categories when classifying data due to URL and similar functions , it becomes more specified and easily dealing with as it reduced from 221 category to 28 only fig.(9)



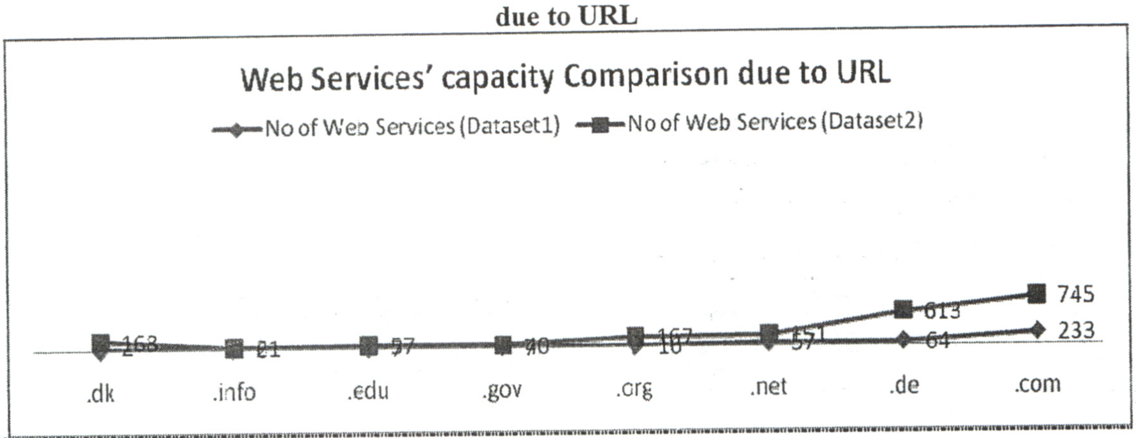
Figure(9)Transfering Dataset2 into a database after classification

Table(2) displays Web Services' Capacity due to URL , first column represents the output from the experimental results using two different datasets (dataset1 which contains 365 different web services , and the second column using dataset2 contains 2500 different Web Services),the third one shows the percentage of the differential between them.

URL	*No of Web Services (Dataset1)*	*No of Web Services (Dataset2)*	*Percentage (%)*
.com	233	745	3.19
.de	64	613	9.58
.net	57	151	2.65
.org	16	167	10.44
.gov	7	40	5.7
.edu	7	57	8.14
.info	6	21	3.5
.dk	4	168	42

Table(2)Web Services' capacity comparison

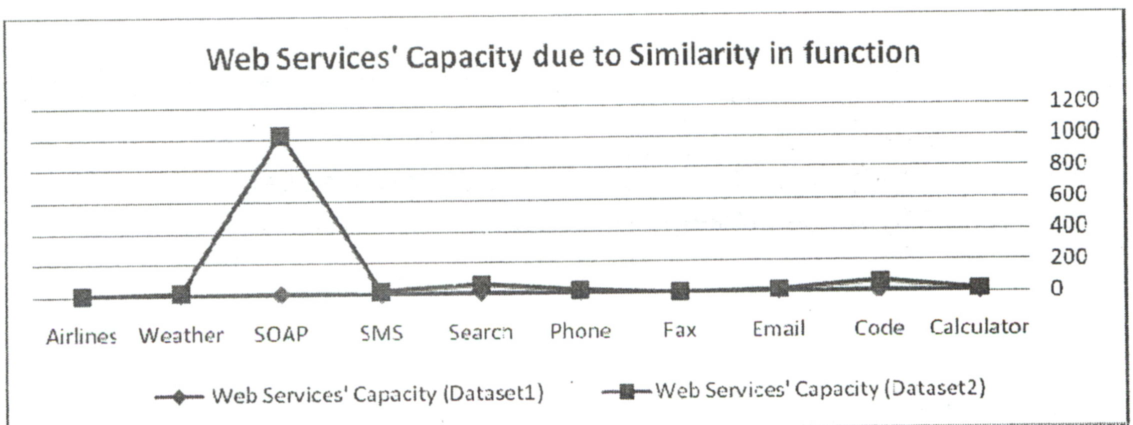
due to URL



Figure(10)Changing in Web Services' Capacity due to URL

Function	Web Services' Capacity (Dataset1)	Web Services' Capacity (Dataset2)	Percentage %
Calculator	3	13	4.33
Code	12	57	4.75
Email	4	13	3.25
Fax	3	3	0
Phone	7	16	2.29
Search	11	56	5.1
SMS	6	23	3.83
SOAP	11	1018	92.6
Weather	6	14	2.33
Airlines	2	2	0

Table(3)No of Web Services capacity due to their function



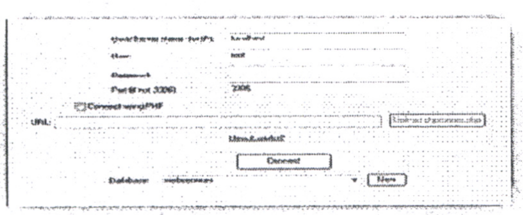
Figure(11)Web Services capacity due to similarity in function

4.2 A Graphical Web Based Interface(GUI)

4.2 A Graphical Web Based Interface(GUI)

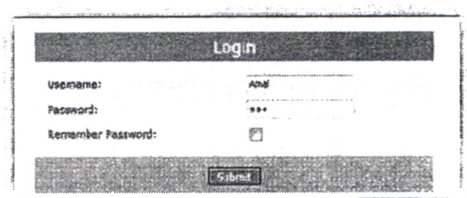
Query forms presented on HTML pages are the only interfaces that users can access the content hidden in online databases, and therefore they are also called deep web or hidden web. as we assumed before that the future image of Web Services' Discovery will be through the web depending on the model of authors at[6] ,we constructed a web site that will be an interface between client and web services' databases , the web site was designed using PHP Runner 5.2 PHPRunner 5.2 and data accessed by MySQL webservices at localhost using HTTP Apache server.

Figure(12)The connection process to access the database



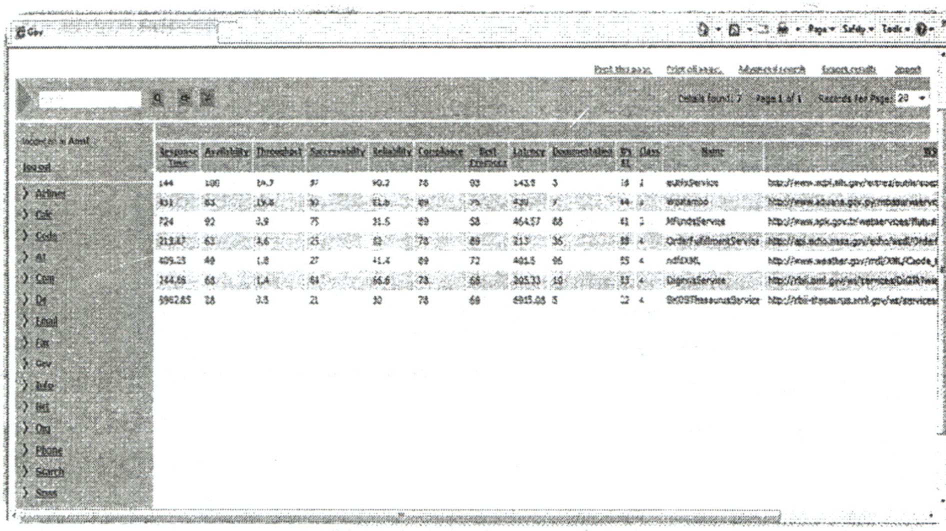
Fig(12)represents the connection process to access the database containing Web Services information

An important issue on the web sites as interface for Web Services' Discovery is the security to protect the access to a resource with a username and a password to authenticate clients by designing a login page fig(13):



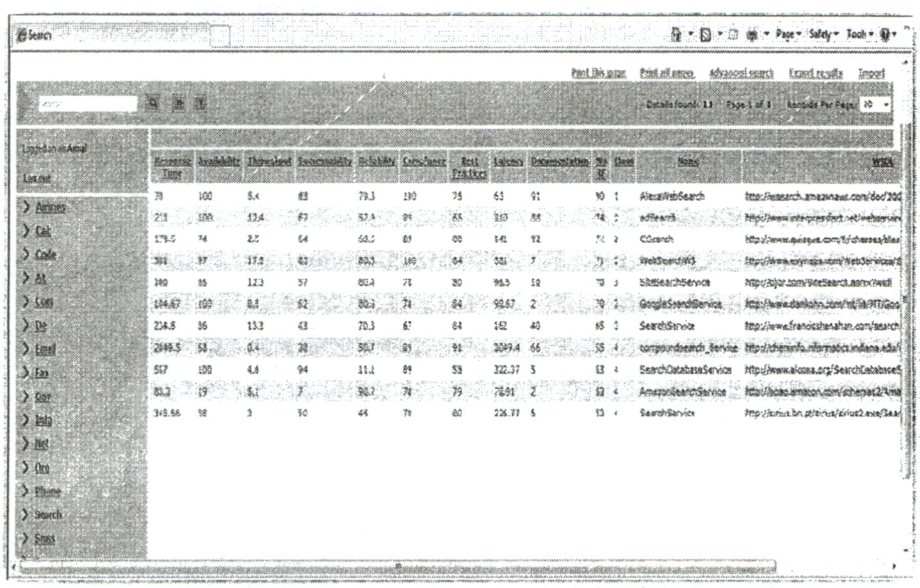
Figure(13)Login Page

After logging in the client will be able to access the GUI fig.(14) which will enable him/her to display Web Services due to URL or due to the function of the Web Service as follow:



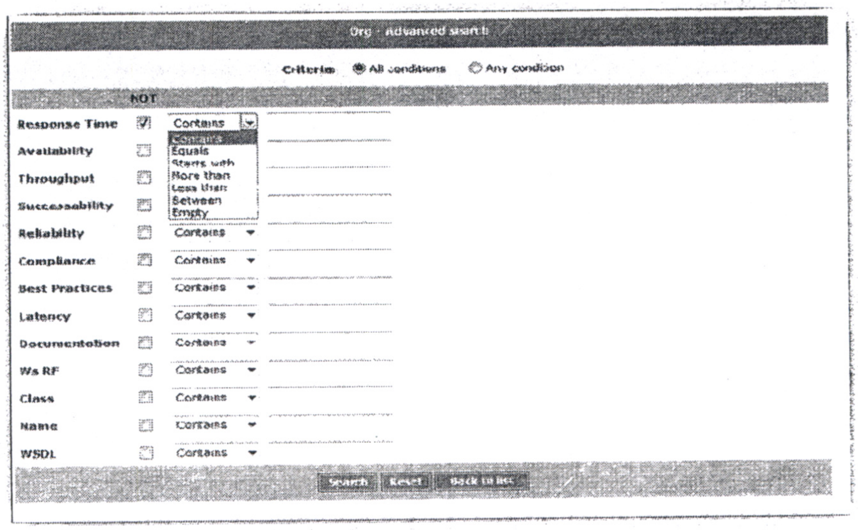
Figure(14)The output from Gov URL Search

For example when client asked for “search” Web Service , he/she can easily select search from the list and the result will be like this:



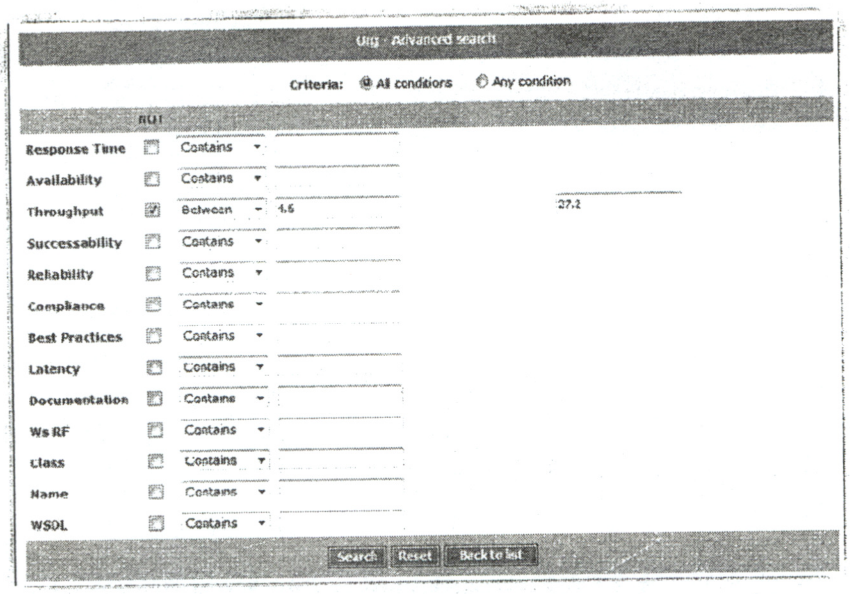
Figure(15)The output from search Web Service

Which shows that there are eleven Web Services for “search” and now it is the role of QoS to select between the different Web Services that have the same functions. Advanced search criteria is applied to the web site ,client can easily enters his/her QoS requirements easily as shown fig.(16):



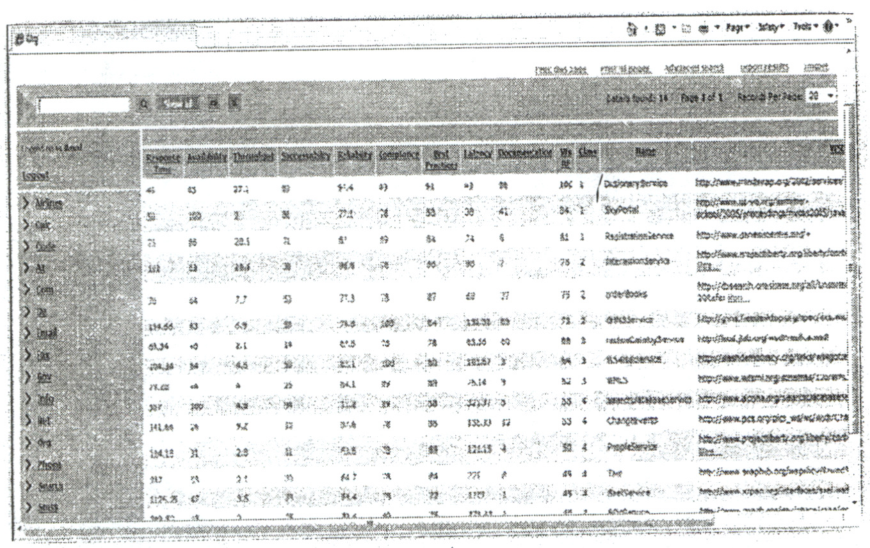
Figure(16)The advanced search for QoS

Figure(17) shows the output from the advanced search which client can use after having different services with a similar function and he/she has to select between them due to QoS requirements.Using interface in fig.(17) Web Service consumer could enter restricted values of QoS parameters to get the Web Service that fulfills his/her requirements.



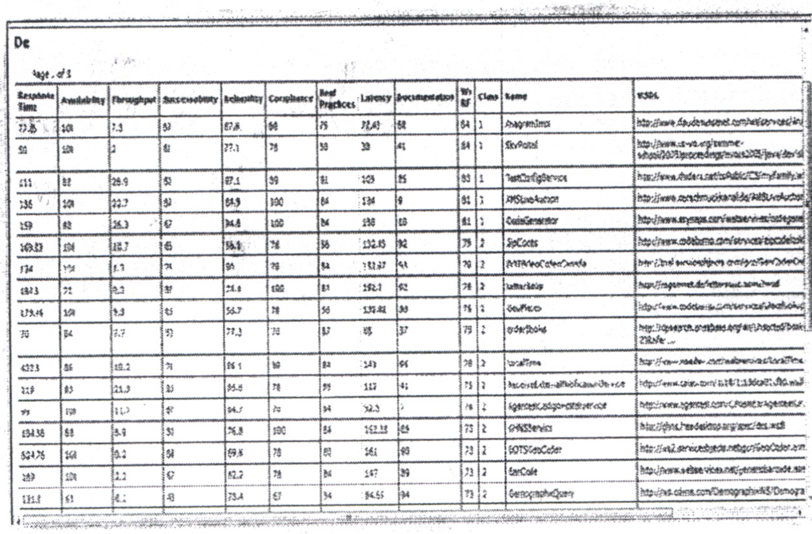
Figure(17)The advanced search after selecting QoS values

And the output will be like this:



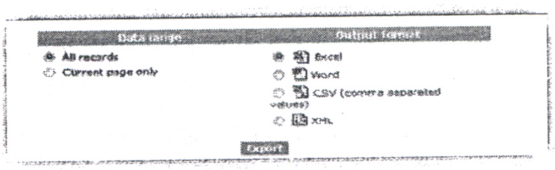
Figure(18)The output from the advanced search

Print the page criteria is also applied to the web site as seen in fig.(19)



Figure(19)Print the page criteria

Data could be exported fig(20) to other programs like word, excel and other



Figure(20)the Export criteria

5. CONCLUSION

From UDDI to Search Engine in the Web Services' Discovery process and merging between them is the future image. Classifying Web Services due to URLs they are published in and grouping them due to similarity in their functions before storing them in WSS could enhance Web Services discovery process .Constructing a web based interface (a client GUI) that could be used by the web services consumer to select between Web Services with the same function with respect to their QoS parameters will facilitate the Web Services' Discovery.

6. FUTURE WORK

After Classifying process a new structure of database depending on the relational entities and building an infrastructure for controlling the storage and retrieval processes will be implemented. Searching for Web Services[20] based on QoS parameters, schema properties, service reputation, trust, and semantic matching will considerably increase the relevancy of finding and selecting appropriate Web services , Some essential QoS parameters were missed from the dataset like cost ,which must be added to it ,studying the ability of frequently updating QoS values of web services must be studied in a future work. at the other side for consumer constructing Open Web Directories contain all published Web Services and their quality of services (QoS) will enhance the WS discovery process, and facilitate it for the consumer to easily fulfill his requirements.

7. ACKNOWLEDGEMENTS

I take this opportunity to express my deep gratitude and regards to Professor Dr. Hesham Arafat and Dr. Ahmed Saleh my thesis supervisors , for their incomparable guidance, supervision and constant encouragement through the course of this research .I would also like to thank Dr. Eyhab AL_Masry for his advices and providing me with datasets for the experimental work.

REFERENCES

[1] **W3C, 2004a. Web Service Description Language.** <http://www.w3.org/TR/wsdl>.

[2] **UDDI, 2004. UDDI Version 3.0.2.** <http://uddi.org/pubs/uddi-v3.0.2-041019.htm>.

[3] ***http://dev2dev.bea.com/technologies /webservices/BPEL4WS.jsp***

[4] http://www.service-architecture.com

[5] **Prentice Hall Service-Oriented Computing** Series *by Thomas Erl.*

[6] http://www.whatissoa.com

[7] **SOA Principles of Web Services Design--** *by Thomas Erl.*

[8] **Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI)** *by Qusay H. Mahmoud, April 2005.*

[9] **Efficient Web Service Discovery and Composition using Constraint Logic Programming** *Srividya Kona, Ajay Bansal, Gopal Gupta1 and Thomas D. Hite2 1 Department of Computer Science The University of Texas at Dallas 2 Metallect Corp.*

[10] **A Hybrid Approach to QoS-Aware Web Service Classification and Recommendation** ***Alexandra Moraru1,*** *Carolina Fortuna Radu Rzvan Slvescu* Computer Science Department, Technical University of Cluj-Napoca, George Baritiu 26-28, 400027 Cluj- Napoca, Romania Department of Communication Systems Department of Knowledge Technologies.

[11] E. AL-Masri, and Q. H. Mahmoud, **"Discovering the best web service"**, (poster) *16th International Conference on World Wide Web (WWW)*, 2007, pp. 1257-1258.

[12] E. Al-Masri, and Q. H. Mahmoud," **QoS-based Discovery and Ranking of Web Services"**, *IEEE 16th International Conference on Computer Communications and Networks (ICCCN),* 2007, pp. 529-534.

[13] **A QoS-aware Method for Web Services Discovery Bian WU, Xincai WU1,2***1Research Center for GIS Software and Application Engineering, Ministry of Education, Wuhan, China 2School of Earth Sciences and Resources, China University of Geosciences, Beijing, China.*

[14] **Model for Web Services Discovery with QoS** *by Shuping Ran 2003.*

[15] *Adam, C., From Web Services to SOA and Everything in Between:* ***The Journey Begins, Webservices.org, May 2005.***

[16] *Reputation-Enhanced Web Services Discovery with QoS.* **By Ziqiang Xu (2006).**

[17] *Discovering Web Services in Search Engines. Eyhab Al-Masri and Qusay H. Mahmoud •* ***University of Guelph.***

[18] *A QoS-aware Selection Model for Semantic Web Services* **Xia Wang1 Tomas Vitvar1 Mick Kerrigan2 and Ioan Toma2.**

[19] *Towards Semantic Web Services Discovery with QoS Support using Specific Ontologies*

Haihua Li, Xiaoyong Du, Xuan Tian.

[20] *Investigating Web Services on the World Wide Web* **Eyhab Al-Masri and Qusay H. Mahmoud**(2008)

[21] *A QoS-aware Model for Web Services Discovery* **Gang YE, Chanle WU, Jun YUE, Shi CHENG, Chanle WU**

[22] *WSCE: A Crawler Engine for Large-Scale Discovery of Web Services* **Eyhab Al-Masri and Qusay H. Mahmoud.**

[23] *A Framework for Efficient Discovery of Web Services across Heterogeneous* Registries **Eyhab Al-Masri and Qusay H. Mahmoud**.

[24] *Google vs. Yahoo* ***Barry Schwartz.***

[25] ***Investigating Web Services on the World Wide Web*** *Eyhab Al-Masri and Qusay H.Mahmoud(2008)*

[26] ***Model for Web Services Discovery with QoS*** *by Shuping Ran 2003.*

[27] ***A QoS-aware Method for Web Services Discovery*** *Bian WU, Xincai WU1,21Research Center for GIS Software and Application Engineering, Ministry of Education, Wuhan, China 2School of Earth Sciences and Resources, China University of Geosciences, Beijing, China.*

[28] ***Towards Semantic Web Services Discovery with QoS Support using Specific Ontologies*** *Haihua* Li, Xiaoyong Du, Xuan Tian.

[29] ***WSCE: A Crawler Engine for Large-Scale Discovery of Web Services*** *Eyhab Al-Masri and Qusay H. Mahmoud Department of Computing and Information Science University of Guelph, Guelph, Ontario, Canadaealmasri,qmahmoud}@uoguelph.ca*

[30] ***J. Day (2004): Selecting the best web service.*** *In: Proceedings of the conference of the Centre for Advanced Studies on Collaborative research, Markham, Ontario, Canada, pp. 293-307.*

[31] ***Semantics in Service Discovery and QoS measurement*** Chen Zhou, Liang-Tien Chia, and Bu-Sung Lee.

[32] ***OASIS: Web Services Quality Model (WSQM)*** TC, http://www.oasis-open.org/committees/download.php/15910/WSQM-ver-2.0.doc.

[33] **Using Common Sense Reasoning to Enable the Semantic Web** *Alexander Faaborg , Sakda Chaiworawitkul , Henry Lieberman, MIT Media Lab.*