A Review on Framework and Quality of Service Based Web Services Discovery

Mustafa Almahdi Algaet

Department of System and Computer Communication
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka, Durian Tunggal
Melaka 76100, Malaysia
E-mail: elget_765@yahoo.com
Tel: +6063316663, +60176594027; Fax: 6063316500

Zul Azri Bin Muhamad Noh

Department of System and Computer Communication
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka, Durian Tunggal, Melaka 76100

Abdul Samad Shibghatullah

Department of System and Computer Communication
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka, Durian Tunggal, Melaka 76100

Ali Ahmad Milad

Department of System and Computer Communication
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka, Durian Tunggal, Melaka 76100

Abstract

Selection of Web services (WSs) is one of the most important steps in the application of different types of WSs such as WS composition systems and the Universal Description, Discovery, and Integration (UDDI) registries. The more available these WSs on the Internet are, the wider the number of these services whose functions match the various service requests is. Selecting WSs with higher quality largely depends on the quality of service (QoS) since it plays a significant role in selecting such services. In achieving this selection of the best WSs, the potential WSs are ranked according to the user's necessities on service quality. In many cases, the value of QoS ontology is realized by its support for nonfunctional features of WSs. This ontology is also capable of providing solutions to the interoperability of QoS description. Moreover, based on the QoS ontology, it becomes more possible to develop a framework of semantic WS discovery. The framework enhances the automatic discovery of WSs and can improve the users' efficiency in finding the best web services. Thus, Web Services are software functionalities publish and accessible through the Internet. Different protocols and web mechanism have been defined to access these Services.

Web Service

Web service (WS) is defined as a core technological device through which users share various resources of knowledge and information and integrate processes in companies or organizations (R. Dinesh Kumar *et al* (2011). Due to the current speedy increase in the applications of this technology, the quality of web services is assuming higher importance as a critical factor in determining the performance of IT system. Moreover, selection of web services is regarded as an essential process in composing web services since it assists users to choose those services with higher quality, thus, meeting the requirements of clients. This paper presents a discussion of a proposed comprehensive WS selection model using Analytic Hierarchy Process. This model is proposed to be used for selecting the best web service based on quality of service (QoS) Constraints. The QoS manager carries out the function of actuating the exact agent for service providers and clients so that they can be involved in processes of performing, publishing and finding web service operations more effectively. Optimizing and ranking the QoS attributes of a web service such as response time, throughput, reliability, availability and cost will be achieved using the Analytic Hierarchy Process.

The Universal Description, Discovery, and Integration Registry

According to Z. Xu, P. Martin *et al* (2007), the most prevailing mechanism of web service discovery is the Universal Description, Discovery, and Integration (UDDI). The registry of the UDDI is regarded as a directory for storage of web service related information. A service provider makes offers users with the opportunity to publish this information concerning the service in a UDDI registry. Thus, the process of locating these services can be achieved by individuals and businesses through searching public and private registries. Airlines, for instance, can publish their fare services to a UDDI registry. Travel agencies then use the UDDI registry to locate Web services afforded by several airlines, and to discuss the service that best meets their needs.

As pointed out by T. Rajendran *et al* (2009), since web service technology and the potential of dynamic service discovery and integration have gained an increasing popularity, it is becoming more possible for multiple service providers to provide users with similar services. Consumers are, therefore, concerned about the service quality and the required functional properties as well. We have recently proposed a simple novel approach to afford QoS-based service discovery. Our model depends on previously existing Web service technology. QoS information which is published by the service providers in the Model structure of the UDDI is used with a reputation manager in order to achieve an authentic QoS-based service discovery. The function of the discovery agent is to assist users to find services that meet the functional and QoS requirements which are specified by the consumers. Assuming that the consumers provide non-malicious and mostly accurate QoS ratings to the reputation manager, ranking these matched services is performed in accordance to the scores of their reputation generated by the reputation manager and the values of their non-functional QoS attributes. Following this is returning the top ranked services to the service consumers. In this case, it is more likely that those services with high, but inaccurate QoS values need to be filtered out by their low reputation scores. Thus, the current paper introduces an algorithm which carries out effective service matching, ranking and selection of web services, and it provides evidence of its effectiveness by discussing a set of simulation experiments.

Figure 1: Model of Reputation-enhanced Web Services Discovery with QoS

K. Kritikos *et al* (2007) stated that the success of the WS paradigm has resulted into proliferating the available WSs. Currently; the available WS standard technologies include advertising static functional descriptions of WSs in the UDDI registries, thus, creating a WS discovery process that does not approve those many irrelevant or incomplete results. Whereas the core aim of inventing or developing semantic functional discovery approaches has been to provide more effective practical solutions to the previously stated issue or problem, the amount of functionally equivalent WS advertisements returned is still large. Such a potential solution to this problem can be achieved in four steps: a) describing the Quality of Service (QoS) aspect of WSs and that is directly relevant to their performance; b) filtering the WS functional discovery results depending on users' constraints on their QoS descriptions; c) sorting the results based on users' weights on QoS metrics. Thus, in its basic definition, QoS of a WS is known as a set of non-functional attributes that greatly affect the quality of the services provided by the WS. Furthermore, measuring each QoS attribute can be obtained by one or more QoS metrics, which provide specifications of the measurement method, schedule, unit, value range and other measurement details. This specification of WS by QoS is materialized as a set of constraints on a certain set of QoS metrics. These constraints limit the values of the metrics in a particularly determined range or in a certain enumeration of values. It is only in the expressiveness of these constraints where the current modeling efforts of the QoS specifications differ. However, these efforts are proved to be ineffective in QoS metric modeling. This is mainly due to the nature of their QoS metric model as being syntactic, poor and not extensible. In this way, the most prominent QoS-based WS discovery algorithms produce irrelevant or incomplete results.

In proposing the QoS ontology, Y. Baocai *et al* (2010) intended to achieve support of the semantic matching of WSs. This QoS ontology provides a formal semantic description of web services non-functional properties. Thus, reasoning on QoS ontology will be an important step in the process of selection of WSs. It is also capable of determining the suitability of the service for the requester's requirements. Furthermore, the traditional UDDI lacks essential descriptions of QoS. Therefore, it is challenging to achieve ranking of similar services only by its efficient functionality. Thus, in attempting to overcome such challenges, it was suggested by some previous researchers to incorporate the QoS information into service discovery process. This is because such QoS related information especially information regarding the reliability and accessibility of a given WS can be well utilized in enhancing the availability and efficiency of WS selection. However, the remaining challenge is the inadequate syntactic descriptions of QoS due to the use of various concepts, scales and measurements among services providers and requesters. Therefore, this makes semantics of QoS necessary in WS discovery for satisfying users' needs.

A. Web Service (WS)

Lu Li *et al* (2010) provided an explanation of the fundamental framework of WSs. This framework consists of three participants and three basic operations. The three identified participants are service providers, service requester and service agent. For the three basic operations, they are releasing, binding and searching. The framework also illustrates the basic workflow. First, service providers will provide services to the service agent on the release. In cases when the service requester has to call the service, the first step it takes is visiting the service agent and obtaining information on how to call the service. This is followed by direct communication with the service provider directly, and calling the required services. Middleware is defined as computer software that serves as a connector of software components or applications, It shields heterogeneous and complexity under the distributed environment. It is also widely used in web applications based on the Web for completing the interaction between different components. (MOM) is a kind of middleware which functions as message transmitter, and it is capable of providing transparent communication services for web applications based on the Web.

B. Quality of Service (QoS)

According to the definition of QoS offered by Lu Li *et al* (2010), it is initially defined by CCITT: QoS is an integrated indicator which is used for measuring satisfaction with a service, and for describing some features characterizing the performance of a service. There has been a general agreement among academics about the narrow and broad sense of QoS. While the narrow QoS refers to some technical indicators of the network transmission, the broad QoS refers to end-to-end QoS, including the network layer, system layer, middleware layer, application layer, including resource allocation and utilization, and consultations between the layers.

This paper providing the specifications of WSs as desired by a client's requirements is an onerous task as there are many WSs which can satisfy a request. The process of WS selection still faces two major issues: recommending the pertinent WSs and lack of providing the unwarranted service. Moreover, the defect of keyword search is that it does not have accurate articulation of the client's request. In other words, most current search engines and UDDI are based on a keyword search, which is inherently impractical for affording the client with the adequate service. Because of the inability of the keyword search to grasp the underlying semantics of the searched query, it is important to introduce the discovery of WSs which relies on the input and output operations. Although this search-based operation returns relevant result, the rank of the latter is not based on performance of WSs. Therefore, incorporating the search mechanism with the QoS requirements is necessary because this is expected to enhance competition of the selection process. This necessitates the need for introducing QoS WS discovery which is based on input and output operation as a new approach in this paper. According to this approach, QoS consultant functions as a broker between the client and service the provider. Moreover, in this approach, the QoS attributes are normalized, and the match score value is stored in the Service pool. The Web Service Description Language (WSDL) Parser carries out the functions of extracting the input/output operation from the WSDL file and storing it in the service pool. In cases when a search is carried out for a given input/output operation, the consultant chooses a list of candidate services that match up properly with the given request that will be provided to the client for setting weights over QoS attributes. Thus, based on this operation, the highest degree of the matched WSs will be provided to the client for invocation process. The system has been tested with real and synthetic data, which shows propitious result.

This study addressed the problem of QoS-based WS composition. It identifies and utilizes the skyline services in terms of their QoS values. Moreover, an algorithm that enhances the effectiveness of the state-of-the-art solution by pruning non-skyline services is proposed and introduced in this paper. In dealing with cases in which the size of the skyline appears to be larger than it is in the initial dataset, a method for selecting and using representative skyline services for the composition was utilized. Another efficient method was also employed in this study to determine the local quality levels,

and hence, enhancing the success rate of the hybrid solution for the QoS-based WS composition from our previous work. Last but not least, we set up a method that is capable of making service providers more competitive in potential client attraction. Thus, based on the results of the experimental evaluation in this study, a more significant performance was gained by this proposed approach as compared to the performance gained by other existing approaches, which depend on global optimization.

Moreover, these results of the present experiments proved that the challenging difficulty that affected the performance of our skyline-based methods is the composition problem, and more specifically, the number and strength of the specified end-to-end QoS constraints. To overcome this obstacle, in the coming work, the researchers managed to develop another effective method through which they could estimate the difficulty of each composition problem. Then, based on the levels of the difficulty identified, it was possible to decide the level of the skyline representative's tree (or the QoS levels tree) the search is going to start, i.e., lower in the case of stricter constraints, in order to avoid unnecessary iterations and have a smaller increase in the execution time.

In integrating the QoS ontology into WSs, its extensibility and flexibility features proved in various domains and applications makes it capable of facilitating various participants by offering specifications of their QoS offers and demands at different quality levels and fine-grained service levels. This ontology comprises QoS metrics and values which are designed in a way that can support transforming related QoS metrics between each other or computing a QoS metric from others. Furthermore, integration of different QoS attributes like tendency, mandatory, weighting, dynamism, relationship, grouping, etc. into the ontology makes the ontology more able to provide reasons about QoS characteristics and facilitate effective computation of QoS rank for WSs. As promising works, developing an AHP-based QoS ranking algorithm for WSs based on the QoS description specified by our QoS ontology was proposed. Moreover, extension of the UDDI registries is proposed to allow for semantic description of the QoS information of WSs and integration of the AHP-based ranking algorithm into UDDI registries. This will enable service requesters to directly get a ranked list of WSs from the UDDI registries.

The low rate of utilization of WS technology is the major aim of this study and as previously discussed, QoS is regarded one of the major issues contributing factors. Although there are other similar Web Services which are also emerging on the Internet, and competing to offer services, there is still a need to enable such mechanisms to carry out effective discovery and selection of WSs. Therefore, to achieve this, a scalable QoS-aware model was proposed by introducing QoS Broker (including QoS Database, Publish, Lookup, Monitor and Valuate module). There are several merits of this new proposed model including introducing the QoS description from the beginning of the WS, publishing services by providers, no need for modifying the standard UDDI interface and allowing the client side software to transparently plug on. Moreover, the QoS Broker performs as a third credible part that carries out three functions namely; coordinating the measurement, monitoring, and updating QoS Database in time to insure the reliability of the discovered services based on QoS.

Figure 2: Register-find-band model for Web Services discovery
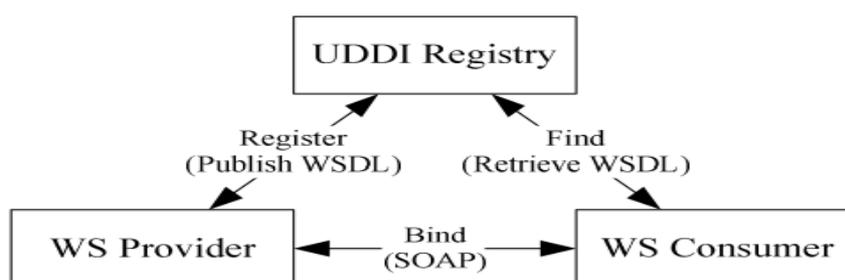
As pointed out by H. Yang *et al* (2009), there is an increasing importance of matching and discovering WSs. Thus, using the domain ontology in describing the semantics of WSs and matching them on the semantic level is becoming a promising area of research. In this study, matching WSs which is considered a problem is transformed to the computation of semantic similarity between concepts in the domain ontology. This semantic similarity is proposed to be measured by from the semantic distance and the factors of Path Length, Depth, Local Density and Number of Down Direction in the algorithm are all taken into consideration. Such a measure is affirmed as an effective means of discriminating services at a level of granularity as it has paved the way to the tremendous progress of the matching process in semantic web services. Due to the speedy increase in the number of the currently existing WSs, the process of discovering more accurate WSs for our needs from numerous WSs becomes the key problem of WS systems. The WSDL provides a description of how to invoke operations mainly, but this still lacks the capability to provide an explicit representation. The UDDI provides a web wide registry of WSs but its syntax-based search granted generates results that are coarse in nature. Thus, its levels of precision and recall do not meet our requirements. This is why the challenge in how to provide accurate semantic information of WSs and how to match semantic WSs become the major focus of the research.

This analysis supporting the discovery of WSs has been paid a great deal of attention by previous researchers because of the proliferation of available WSs and the failure of the syntactic approach of UDDI. Therefore, providing more accurate results has been the major purpose of inventing semantic discovery mechanisms recently. However, many functionally equivalent WSs are returned by semantic WS registries. Fortunately, enforcing the semantic QoS-based description and discovery of WSs has become the main solution to these problems. The researchers introduced a rich and extensible ontology language for describing QoS-based WSs and proposed a semantic QoS metric matching algorithm. Based on this algorithm, the researchers proposed an extension of the CSP-based approach for QoS-based WS discovery. They also showed the extended OWL-Q with SWRL rules as OWL alone fails in some aspects of QoS description. Moreover, they modified the metric matching algorithm to make it more feasible. Following this was proposing two novel semantic QoS-based WS Discovery algorithms capable of returning matches even for over-constrained QoS-based WS requests. The first one deals with unary constraints while the second one is more generic.

The quality of WSs as an aggregated metric for providing a description of the features characterizing the systems in areas such as networks and distributed systems. By using the standard Web Service Distributed Management (WSDM) and WS Ping, this study introduces a newly lightweight approach that is capable of solving the WS QoS problem. The Management of Web Services (MOWS) which is recognized as part of WSDM is used to publish WS's QoS parameters. Management using Web Services (MUWS), the second part of WSDM, is used to monitor IT resources' QoS. Server's QoS, application server's QoS and network's QoS are all examples illustrating this. It is also possible to use WS Ping as a simple diagnostic tool for WS's latency and availability across organizational boundaries. Therefore, the standardized WS Ping operation into all WSs is proposed in this study. While MOWS, MUWS and WS Ping are used for retrieving all QoS data, they can be also utilized for selecting WSs. This study presents a new architecture of WS selection and proposes the Delegation Web Service as a selector. In comparison to other selectors such as Web Service Broker, consumer and QoS enhanced UDDI, the Delegation Web Service as a selector can provide a better solution for the desired implementation of WS load balancing and achieve an increase in the security of and for WSs.

Web Service Ping for Quality of Service

Ping is a defined as a well known simple tool used in checking whether a given resource under a particular IP address is available or no and whether the round trip time (RTT) of an IP based request-replies messages. It also generates well-defined loads to test IP networks. Thus, Ping reflects various desired characteristics that can be an inspiration for a simple diagnostic tool for WSs.

The main focus of the new proposed framework introduced in this paper is twofold: to conduct an analysis of the requirements of a semantically rich QoS-based WSDM and to provide SW and constrained based mechanisms for the purpose of enriching syntactic QoS-based WS Discovery (WSDi) algorithms. Furthermore, the paper presents a roadmap of extending WS standard techniques for realizing semantic, functional and QoS-based WSDi. WSs are also defined as applications of modular, self-describing and loosely coupled software whose advertisements, locations and use are found across the Internet using a set of standards such as SOAP, WSDL and UDDI. They are also characterized by being flexible to be dynamically discovered and integrated at runtime for the purpose of developing and deploying business applications. However, the standard WS techniques such as WSDL and UDDI are proved to be unable to realize dynamic WSDi. This is because they depend on static rather than dynamic descriptions of service interfaces and other non-functional service attributes for publishing and finding WSs. The result of this is that syntactic WSDi mechanisms are characterized by returning results with low precision and recall. Moreover, no means are provided for selecting among multiple functionally equivalent WSs. This paper provides an analysis of the requirements for semantic QoS-based WSDD and a description of proposing extension of the current standard WS technology for the purpose of incorporating and using an appropriate semantic QoS-based WS framework needed to be supplied. An analysis of the responsibilities of the participating entities in the WSDi architecture was also performed in sight of the incorporation of the referred QoS-based framework. Moreover, the current focus of this study is accomplishing the specifications of OWL-Q and the implementation of our semantic QMM algorithm which have been previously discussed. In extending this work, it would be possible to achieve a QoS-based WS composition through enforcement of global QoS constraints and usage of every QoS metric, metric evaluation functions which are imposed on any possible workflow (WS) composition construct, thus, producing global QoS metric values by reducing the constructs.

The proposed QoS based WS Selection Model (WSSM-Q) which aims at providing the QoS support for service publishing and selection in this study includes managing the QoS of Web services such as defining the QoS model, collection of the QoS information, computation and maintenance of the QoS data. Thus, based on the QoS management, the WSs that submit the needs of the consumers are ranked for selection according to the overall utility of the QoS. Following this is carrying out the implementation and evaluation of the prototype system through experiments. WSs are also essentially recognized as distributed applications that undergo universal deployment and they can be invoked using standard protocols. Due to the increasing number of functional similar WSs, the process of selecting the best WSs which can satisfy the consumers' requirements represents a new direction of an ongoing research in WS community. However, this WS selection is usually dedicated by functional requirements that cannot guarantee the real-time validity of these WSs being selected.

The Proposed Framework for WS Selection

A typical WS model (the right part of this Fig 3) includes three entities namely; the service provider, the service requester and the service registry. For service discovery, some requirements concerning the capability of a requested service are provided by a service requester. In addition, a wide number of service providers advertise the capabilities of their services as a way of publishing them. In the process of any specific task accomplishment, the service requester engages in two tasks; interacting with the service registry to discover potential services and invoking the most appropriate one. In such a case, the discovery process which has conducted the process of matchmaking the nonfunctional and functional properties of services produces several similar services. Thus, based on the result obtained from the WS selection process, one of these similar services will be selected which is illustrated in left part of this Fig.

Figure 3: The framework of Web service selection process
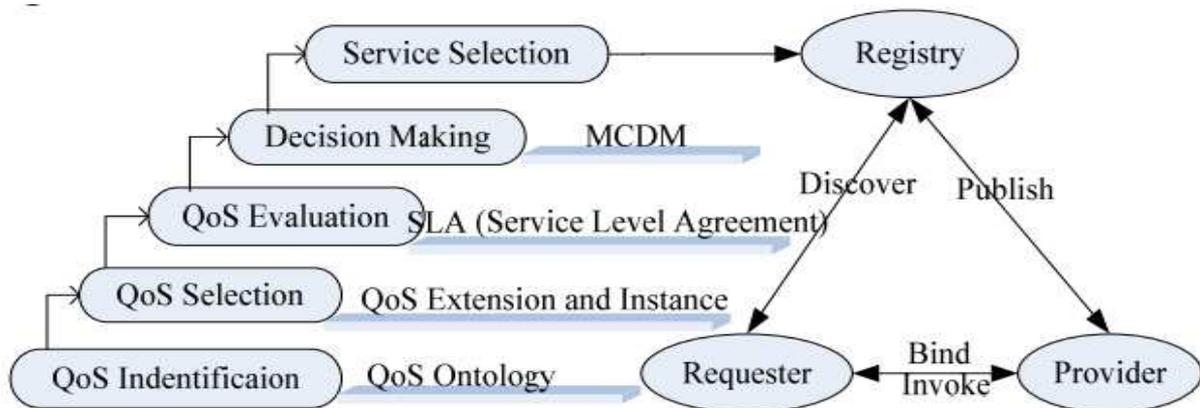
A wide number of studies have been carried out on the existing WS QoS representation. However, in addressing their inadequate representation of aggregated QoS attributes and values as well as lack of consistency checking, this current research proposed three different approaches to modeling QoS models in the UDDI. The challenging issue encountered in achieving an efficient QoS model composition in WS relies on several aspects including consistent representation of the QoS model, efficient acquisition of QoS values and storage of composite QoS values. Based on the results of the experiments, the performance of the proposed type and keyword based approaches in retrieving and storing QoS values is better than the performance of the other approaches. Moreover, the performance of these proposed approaches in locating the required service is even more efficient than the performance of other approaches. Regarding the proposed ontological approach, it seems less effective than the above proposed approaches because of the burden of processing time on the OWL file. However, it shows better performance than the other existing ontological approaches. Therefore, the future work of this research is identifying the QoS composition patterns which can hopefully facilitate QoS representation and QoS value aggregation in the service composition process.

Provisioning services across multiple or heterogeneous registries, differentiating between services that share similar functionalities, enhancing end-to-end QoS, and enabling clients to customize the discovery process are the main challenges encountered in in WS discovery. Due to the Proliferation and interoperability of this multitude of WSs, new standards on how services can be published, discovered, or used (i.e. UDDI, WSDL, and SOAP) have emerged. Such standards can provide many of these potential characteristics and even much more, but there are still several technical challenges related to these existing standards. One of these major challenges is related to the client's ability to control the discovery process across accessible service registries for finding services of interest. This study suggests a solution to overcome this impediment by introducing the Web Service Relevancy Function (WSRF) which can be used to measure the extent or degree to which the ranking of a specific WS is relevant based on QoS metrics and client preferences. We present experimental validation, results, and analysis of the presented ideas.

The QoS is an essential factor that should be taken into account in service composition and recommendation. Since the static QoS data cannot reflect the real-time performance of a certain WS, a mechanism was required to help collect the dynamic QoS data. Also, the QoS query result depends on the QoS data stored by service registry to be determined. Consequently, it was important to use a practical and accurate feedback method for calculating and transmitting those dynamic QoS data to the service registry and a QoS feedback model was proposed based on the objective QoS metrics using some simple statistical theories and a dynamic queue as a data pool to cache all runtime status. Furthermore, to enable the service provider to assign less hardware resource and to avoid disturbing feedback result from unfriendly exception, the error determination and sampling were taken into account. Thus, the experiments carried out in this study provided evidence of how this feedback model

evaluates the WS performance, and even better than other common methods. This is because this model provides QoS metrics that are easy to rank and sensitive to the status change.

Web service technology and its dynamic service discovery and integration are getting increasingly popular. Therefore, multiple service providers are now supplying similar services. The QoS is a recognized as a decisive factor in distinguishing functionally similar WSs. The proposed theoretical framework is based and implemented on the QoS properties. In developing this framework, it is important to architect the new model and develop an agent-based framework to support the QoS aware discovery of WSs, develop the matching algorithm that exploits syntactic and semantic information from WSDL specifications. An amount of services whereby we can test the performance of our system is highly required. To select the best model, we need to simulate the matching process under different IR models as well.

Figure 4: Web service discovery Framework
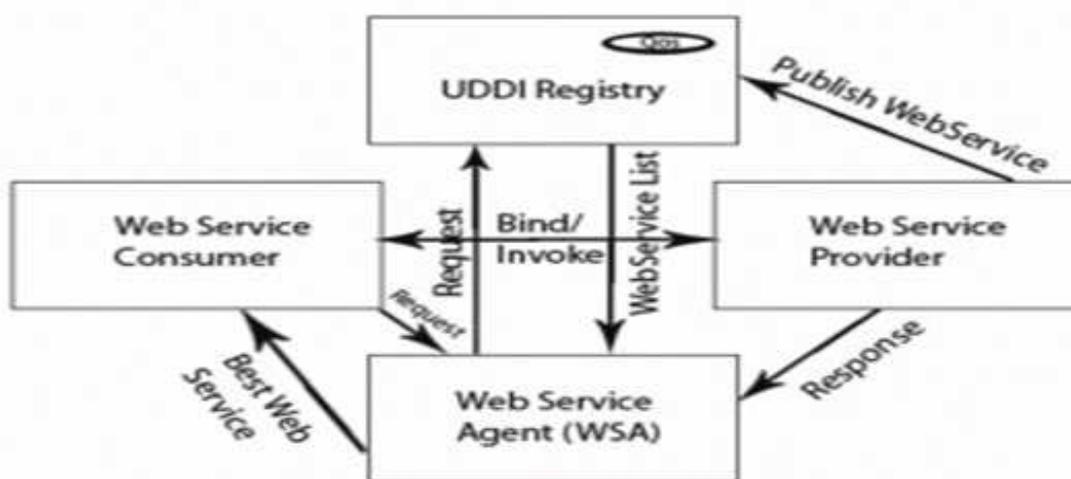
The model can guarantee that the query services results meet the users' functional and non-functional requirements and enhance performance. Regarding enhancement in efficiency, this is realized when users request certain information and especially through division of this request information into two parts, and separation of the QoS and service for parallel processing to avoid dealing with them together and reduce selective time. For improvement in precision, this is achieved through the algorithm for WS discovery which is based on OWL-S description and enhancing the performance of WS discovery in comparison to other existing methods. The query results meet or satisfy the users' precise requirements particularly for the QoS. Concerning the semantic support, the model can settle the request of semantics in the users' services by forwarding the similarity of semantic concepts based on their ontology. However, the decision-making process in selection and combination of WSs depend on multiple sources such as Trust and security.

The QoS selection for WSs is becoming a significant challenge. Therefore, an advanced QoS based selection framework that manages WS quality was suggested and mechanisms for QoS updates have been supplied. The proposed QoS_IC framework allows preserving the current UDDI standard architecture and introduces new components that can be added seamlessly without making any change in the current registry, and it is possible to customize it for specific domain. QoS modes were derived from the assumption that network conditions are static. In practice, network factors would directly affect many QoS properties such as response time. Therefore, this paper further addresses this hindrance in our coming works. Furthermore, the awareness level of WSs was surveyed along with a service registry in UAE to assess the businesses` willingness to adopt and use a service registry.

All these modules support a unified service registry interface for the service providers registered to different UDDI registry. This interface can make publishing the service to the service registry easier. At the same time, it is expected to afford the QoS attribute information that is provided by the service publisher to the QoS processing module, and store classified database into the QoS attributes. Moreover, a WS combination dynamic selection framework was designed and implemented based on the QoS and visual process modeling was supported. The overall structure of the system and operating mechanisms were also described as well. The paper also provides a description of the QoS global optimization algorithm the system involved, and introduces an improved hybrid algorithm. By implementing the editor framework of GEF, a WS process design editor was achieved, and a simple Prototype system was also implemented.

Figure 5: Architecture of the framework
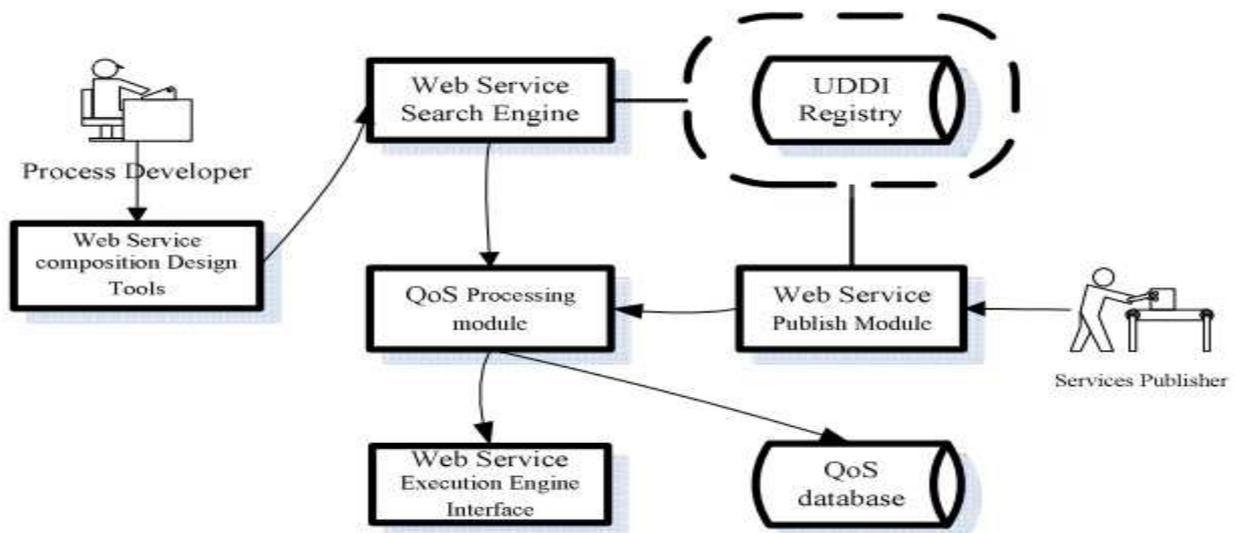
The proposed model for WS discovery by Patil *et al* (2010) is based on the QoS and a high level algorithm for matching and ranking, and selection of services should be implemented by the Web Service Broker. This model is also based on the recent feature in the UDDI registry for storing QoS information of WSs. Publishing the QoS information is performed by the service provider at the time of service registration and can be updated by them at any time. Based on the specified requirements of the service, the QoS requirements (optional), ratings requirements (optional) which are specified by the service consumer, the WS Broker responds with the set of services which meet the corresponding requirements.

The same researchers 'discussion introduced some proposed extensions and highlighted their benefits for WS in the service discovery phase. Moreover, the concepts of Quality Attributes and Quality Weights which enable the service consumers to take a more deterministic and predictable decision in selecting binding to the service were all introduced. Another benefit of these attributes is that they can make the unpredictability of fail-over situations easier and support selection of a backup service during the recovery. These changes were recommended to be implemented in UDDI to assure a true end-to-end QoS capability for WSs.

They also discussed some sub-problems including methods of matching service, evaluation of system discovery and request formats. The QoS properties upon which the proposed framework for WS selection was based were collected implicitly by a distributed agent-based system. Moreover, the researchers intended to apply the recommendation system presented for the WS discovery. For this, implementing a matching algorithm, and enriching the current functionality of agents with the ability to mediate WS invocations and measure QoS factors were two required.

Figure 6: Web service Discovery Model
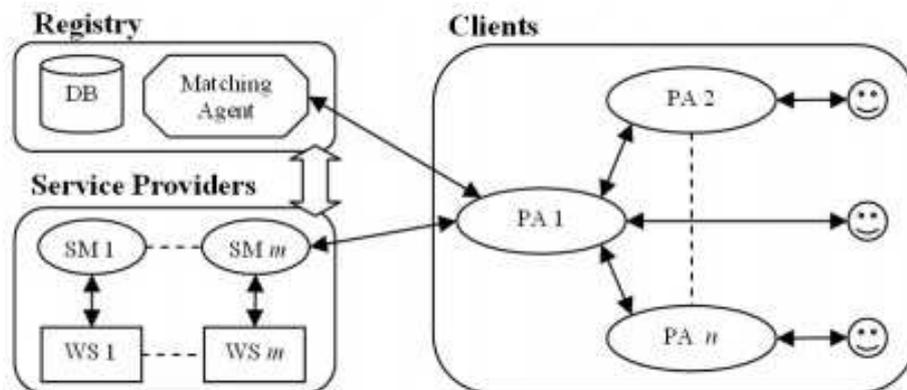
It is stated that personal agent (PA) performs two tasks: accepting the requests from its client and sending them to registry. The matching agent offers lists of relevant services and agents searching for them. PA requests information from agents and service mediators (SMs) for and, they provide it. PA ranks the services and invokes the best one. Both PA and SM store execution statistics.

In a study by Patil *et al*.(2012), several mechanisms for the WS discovery based on the QoS were studied and a number of studies on the existing approaches for storing and representing web service QoS were conducted. The proposed approach to modeling and storing WS QoS parameters and their values in Model of UDDI registry is efficient to selection of WSs from different WSs according to their QoS values. Most of all, having a more efficient QoS Model composition in WSs depends on many factors including consistent representation of the QoS model, efficient acquisition of QoS values and storage of QoS values. Thus, the performance of the proposed keyword based approach in retrieving and storing QoS values is better than that of the other approaches which are the most efficient approaches in locating the requested service.

A great deal of research has been conducted to identify the issues related to the distributed UDDI. Most previous studies recommended combining the distributed architecture and UDDI mechanisms as a promising methodology and suggested a number of alternate solutions. Each of these solutions has its own advantages and disadvantages, and no single solution has been reported to satisfy bandwidth efficiency and query expressiveness requirements simultaneously. Previous research also introduced various distributed computing infrastructures which can be utilized for distributing the functionalities of UDDI. One of these technological infrastructures is the P2P architecture which can be used for effective distribution of the functions of UDDI. Moreover, using a partially decentralized P2P architecture would achieve efficiency and flexibility of the network. We can use the OWL language for describing WSs which facilitate efficient and semantic enabled discovery. As far as decentralizing UDDI is concerned, the multiple copies of the entire directory information are stored at different network locations. This hybrid directory structure provides better bandwidth consumption, scalability, and fault-tolerance. However, all possible combinations of architectures and distributed technologies are still unexplored yet. Hence, in order to realize the promise for an efficient and flexible distributed discovery of web services in large scale distributed UDDI, the coming research in distributed UDDI should focus on unveiling the hiding alternatives.

Figure 7: Service Discovery model Distributed UDDI
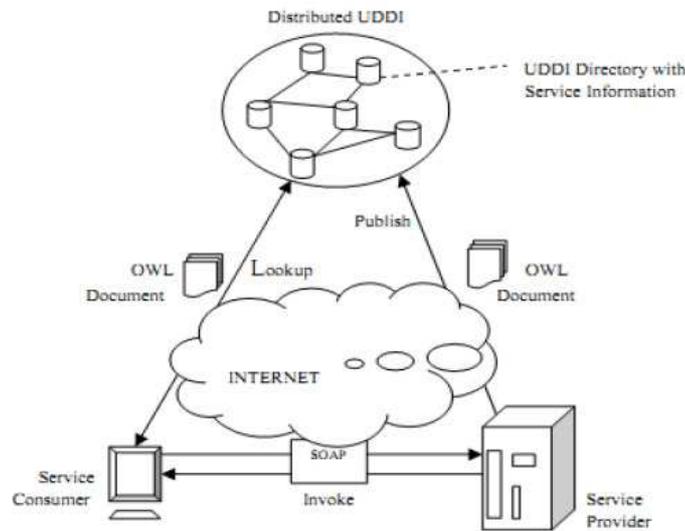
It also introduced two alternative approaches to customizing private UDDI registry query results. This was based on using personalization parameters which could be stored within or outside the service registry. From the conceptual perspectives, these practical approaches are applicable to other SOA registry candidates such as ebXML. In measuring the performance and scalability of each approach, the experiment results revealed that the storing the parameters in external resources is an approach which was found to be more effective than keeping the parameters as keyed reference value within UDDI. The results of the closer investigation showed that what degraded the performance was the time taken by UDDI proxy to obtain various UUID keys before it could retrieve the parameter values from the category bags.

Figure 8: Proposed model architecture
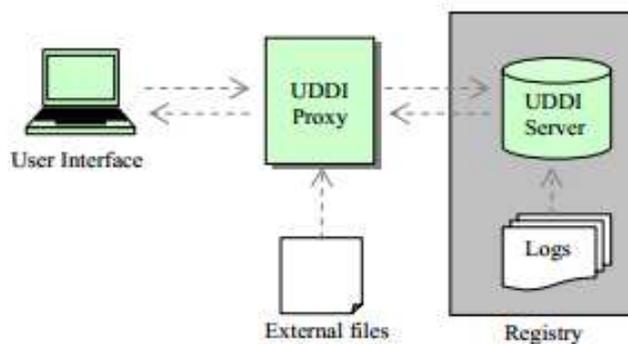
Web Service –Quality of Service Broker Architecture
The WS-QoS broker comprises four components namely; Service Publisher, Verifier and Certifier, Service Selector and Web Service Storage (WSS). It is possible to use such broker services to facilitate service registry access. Besides performing the interaction with the UDDI, the broker also provides the QoS management operations. For the success of the WS technology, the integration of the various QoS properties is mainly essential. Therefore, multiple service providers are now providing similar services due to the increasing popularity of the WS technology and the potential of dynamic service selection and integration. Moreover, the current web service selection is severely hampered by the absence of a mechanism that considers QoS properties for the web service selection. Thus, an approach that reduces

the complexity of matching users' requests according to the specified functional and QoS requirements is currently proposed. Furthermore, a new WS-QoS broker based architecture that solves the problems associated with WS selection is carried out as well. The broker performs the process of publishing and selection of web services. Our suggested theoretical architecture is based on the implemented QoS properties. Eventually, a great deal of services are needed to monitor and assess the system`s performance. And this will assure a more flexible, and reliable architecture

Figure 9: Architecture for WS-QoS Broker
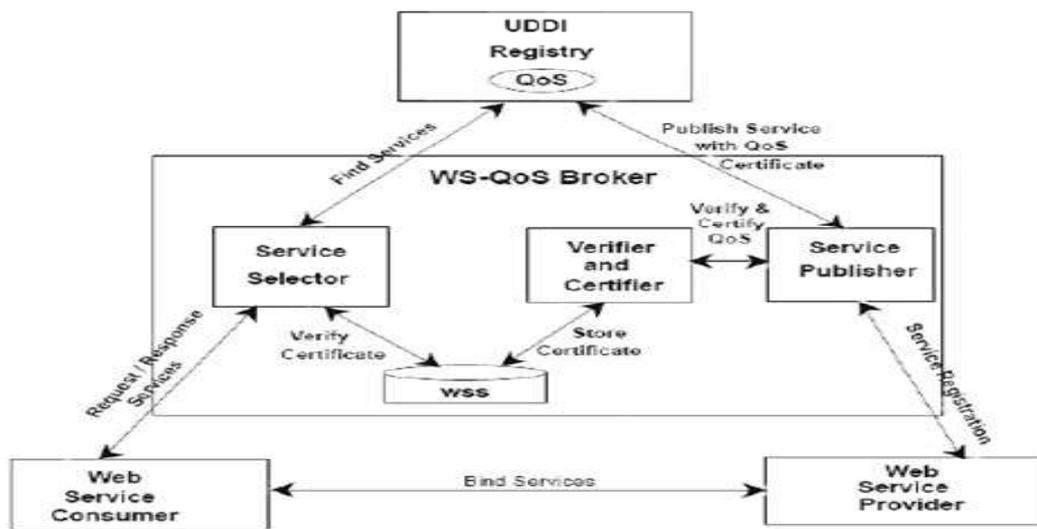
According by P. Bonderud *et al* (2012) this research mainly aims at offering a mechanism that allows trustworthy relationships among service producers and consumers. Generally, the issue of trust is derived from the common practices in which service producers advertise their services through the UDDI. For trusting the reliability of information contained within the UDDI, it is important to validate the authenticity of claims made by both sides, service consumers and producers. From the QoS perspective, it is necessary for the consumer of a given service to have knowledge regarding if this particular service will satisfy his or her requirements and for which various WSs they are contracted. However, from the perspective of security, it is important for the service consumer who obtains services for which they are unable to pay makes a security concern for service producers. Therefore, achieving this requirement of the secure Quality of Services for Web Services (S-QoS4WS) requires a mechanism through which both service consumers and service producers are enabled to establish trust in their prospective partners before involving themselves in any service in which they are required to pay for the QoS. Thus, in attempting to solve this particular challenging issue, this study provides a mechanism that utilizes 'Publisher Assertion' tags within the UDDI to satisfy Security and QoS issues, which are considered out-of-scope for the latest version 3 UDDI release. Using the available architecture for version 3 of the UDDI provides both service consumers and producers with the possibility of establishing a trustworthy relationship in terms of their own interests before engaging in making service transactions. This approach has several benefits including re-using the available UDDI mechanism, establishing trust, and a solution for non-repudiation.

The currently proposed approach to establishing the desired trustworthy QoS in WS selection extends the standard UDDI and presents a new web services registration and discovery model. Moreover, in proposing this approach, a trustworthy QoS prediction model was implemented, and in which the QoS data could be detected and collected at real time, so as to achieve the trustworthy QoS data by T_QoS Algorithm based on PAM clustering. The results generated from the experiments illustrated that the trustworthy computing approach to the QoS of WS selection is effective in terms of

adjusting itself to distortional QoS data and is capable of disposing of the influence of malicious attacks, failures operations and false information.

The QoS in selection for WSs is becoming a significant challenge to undertake. An advanced QoS based selection framework that deals with WS quality and provides mechanisms for QoS updates is suggested here. The proposed QoS_IC framework maintains the current UDDI standard architecture and offers new components that can be added seamlessly without creating any change in the current registry. It can also be customized for applications in other specific domains.

**Figure 10:** Web Service based framework for adaptive workflow management
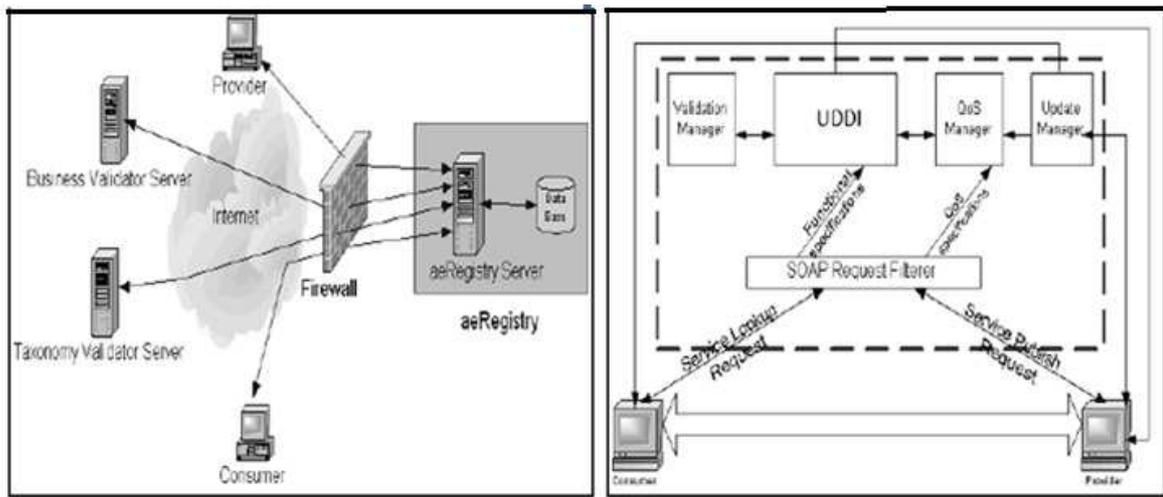
A comprehensive QoS model can afford consistent interface of WSs to workflow-based applications. The incorporated rules in the proposed framework facilitated establishing a dynamic workflow by accomplishing a fine-grained control over service selection. Besides, based on the experimental results, it was evident that the framework is effective in improving the overall QoS of the system. As part of an ongoing work, the rule-base system was extended to attain automatic WS composition using an advanced semantically enriched WS model that allows semantic service matching, composition and choreography.

Figure 11: The Web Architecture
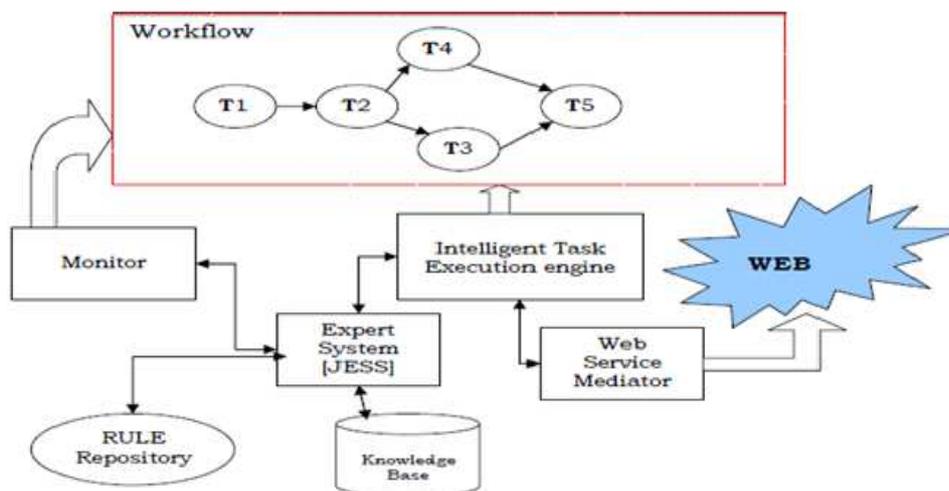
The proposed M. Rathore *et al* ((2011) QoS broker-based process model for dynamic WS composition assures the effective quality of the delivered service and improves the efficiency of composition. The suggested theoretical architecture is based on the QoS properties. The major aim of the QoS broker is to support composite WS discovery with QoS registration, verification, certification and confirmation, optimization, selection and composition. Thus, the process of WS registration and matching by service broker, and the QoS broker key features dealing with QoS for WSs are all described in this study. The model is expected to guarantee the service consumer for *delivering high quality services* and solve other problems such as providing up-to-date services.

Figure 12: The detailed QoS broker based- process model for dynamic WS composition
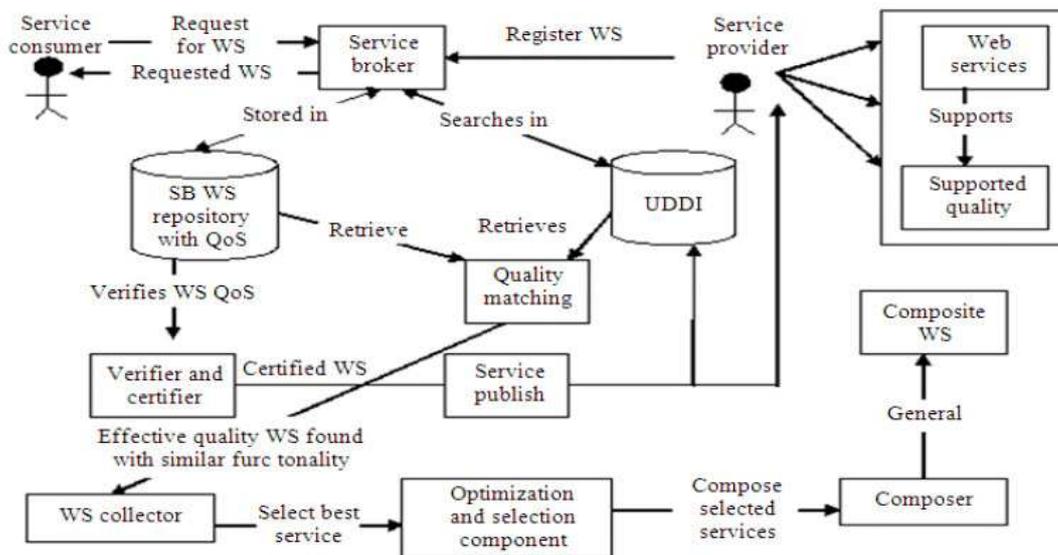
According to Sasikaladevi *et al* (2010), many different WSs are emerging increasingly these days. This makes choosing the quality of WS a crucial but challenging process. Reliability of WSs is regarded as one of the predominant QoS factors in WS selection. The researchers in this study proposed a simple framework model for the selection of reliable WS among the available proxy services. A reliability evaluation framework model was also designed. In carrying out estimation of this, the researchers used availability and accessibility parameters. The values of the probability values are received from the service producers and service consumers. Moreover, simple vector calculation was performed to identify the reliable services.

Figure 13: Framework model for Reliability Evaluation in Composite Web Service
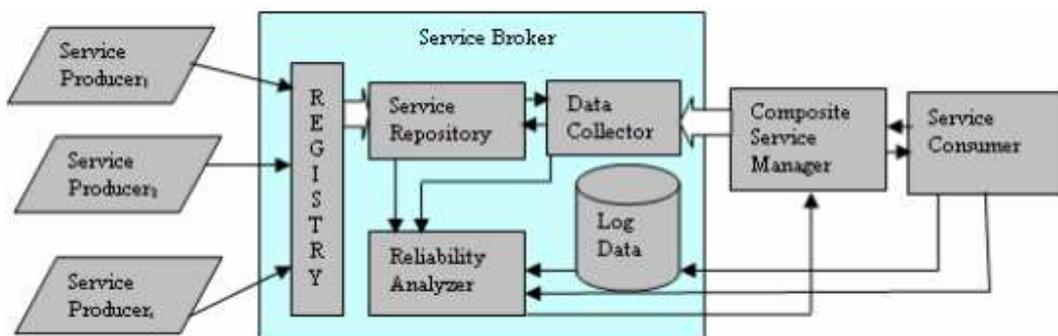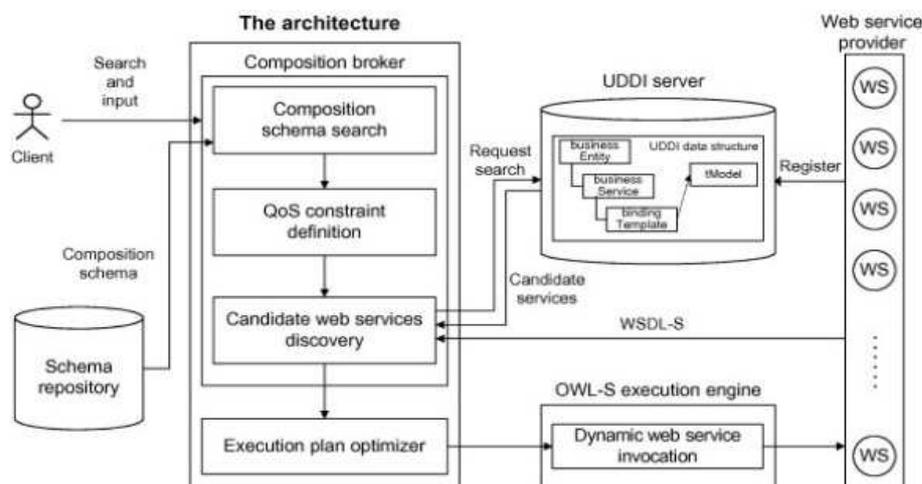
The researchers offered a proposed QoS ontology for expressing QoS information and constraints and an analytic hierarchy process (AHP)-based ranking algorithm for WS selection. This paper contributes to work in this research area in terms of duplicating each one of them as it involves several merits.

First and foremost, the most important advantage of the QoS ontology is its facilitation of different participants' specified QoS offers and QoS demands at different quality levels and fine-grained service levels. The way in which the QoS metrics and values are designed supports transformation of relevant QoS metrics between each other or computing a QoS metric from others. A variety of QoS attributes including tendency, mandatory, weighting, relationship, grouping, and so forth are integrated in the ontology to provide reasons justifying the QoS characteristics and facilitating efficient computation of QoS rank for WSs. Furthermore, the QoS model takes into consideration the variant multiple roles and preference of users and system brokers in specifying their QoS demands. Moreover, this paper places an emphasis on the different important levels between the required QoS constraints and optional QoS constraints.

Secondly, the same researchers proposed an AHP technique for formulating the QoS-based ranking problem and implemented a flexible and dynamic ranking algorithm which is capable of introducing various strategies used in evaluating different applications and domains, depending on the characteristics of the QoS properties when they are applied in those domains. Furthermore, the researchers formulated a number of rules for comparing the QoS value to different QoS value types and QoS tendency. They also attempted to handle the difference in QoS values, units, and measurements with a uniformed mechanism in order to achieve a fair comparison among WSs. The algorithm can also provide support for grouping and prioritizing QoS properties.

In that the researchers presented an effective QoS-oriented web service composition algorithm which has the ability to automatically assign WSs with a high quality to the atomic processes of a composition schema in an optimal way that the six QoS constraints set by the client are not violated. One of the most characteristic that distinguishes this algorithm is that it has a hybrid meta-heuristics that is able to combine both search and simulated annealing and the main of its design is only to figure out constraint–compliant composition plans but also to diminish the computational burden needed for identifying the plans. The movement of the tabu list and probabilistic to the inferior plans makes it possible for the algorithm to find constraint compliant plans very fast. The experiments proved that the algorithm is highly efficient. Moreover, the same researchers designed the QoS-oriented WS composition architecture that assists the clients to choose well-defined composition schemas maintained in the service category and automate the QoS-oriented WS composition. Eventually, QoS based WS composition will be adapted effectively to the concrete world as a best testimony that QoS integrity is preserved at all WS providers.

Figure 14: QoS-oriented web service composition planning architecture

The researchers offered a practical solution to assist versioning WSs. In WS architectures and in SOA, support for versioning is of great importance. WSs do not provide explicit support for versions and all proposed workarounds have considerable disadvantages, however the proposed solution in this paper deals with versioning of WSs at the time of development. Furthermore, the researchers addressed versioning support at run-time. This includes allowing dynamic bindings to specific WS versions. Thus, the solution offered in this paper comprises extensions to WSDL, to UDDI, and to Java annotations. It supports coarse-grained versioning, where a single WSDL or a set of related WSDLs are versioned. At the same time, this solution provides a support for fine-grained versioning in which different parts of WSDL interface, such as types, messages, interface, bindings, or service endpoints are versioned individually. The researchers also proposed an automated approach to enhancing consistency and to simplifying service endpoint mapping for different versions.

Offered a description of a proposed WS discovery model which takes into consideration the functional and nonfunctional requirements during service discovery. The proposed infrastructure comprises a set of services and tools for supporting an integrated WS discovery and solution of selection. The paper also presents a mechanism through which the quality information is carried into either the WSDL description or in a database of a WS selection intermediary (broker-like). Furthermore, it is capable of retrieving these quality features from the WSDL description as well as from the database and processing them. This is to as an attempt to find the services which are characterized by their maximized QoS from all WSs that share the same functional characteristics. By doing this, this work accomplished WS discovery which is based on both the quality and functional characteristics. Moreover, it overcame certain constraints, thus, achieving QoS-based improvements in the WS selection process for elevating the effectiveness of the delivered services.

Figure 15: The general architecture of the proposed model
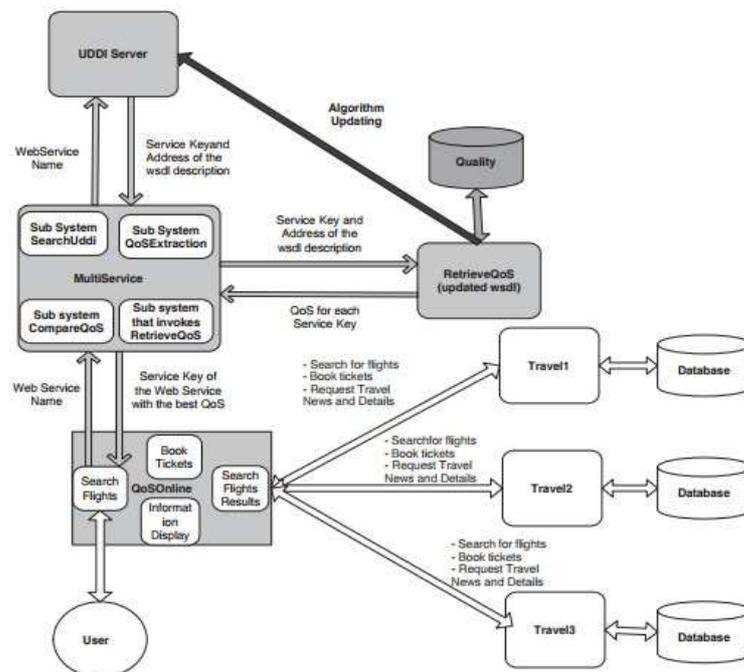
A weighted ontology-based semantic similarity algorithm for WSs, which is able to provide a support for a more automated and veracity service discovery process. In achieving this, it distinguishes among the potentially useful and the likely irrelevant services and it also orders the potentially useful ones based on their relevance to the developer's query.

They propose a fuzzy-based UDDI combined with a QoS support. This work differs from other studies in the sense that the proposed method attempts to take into consideration both the objective factors described by service providers and the subjective information along with an opportunity for users to carry out trust ability evaluations. In this work, the researchers adopted a genetic algorithm (GA) to obtain information concerning users' preferences, and they applied a fuzzy logic for decision-making processes. This fuzzy query interface to input subjective and objective factors enables users to identify the most appropriate WSs for their personal use.

Figure 16: Web service with QoS agent architecture
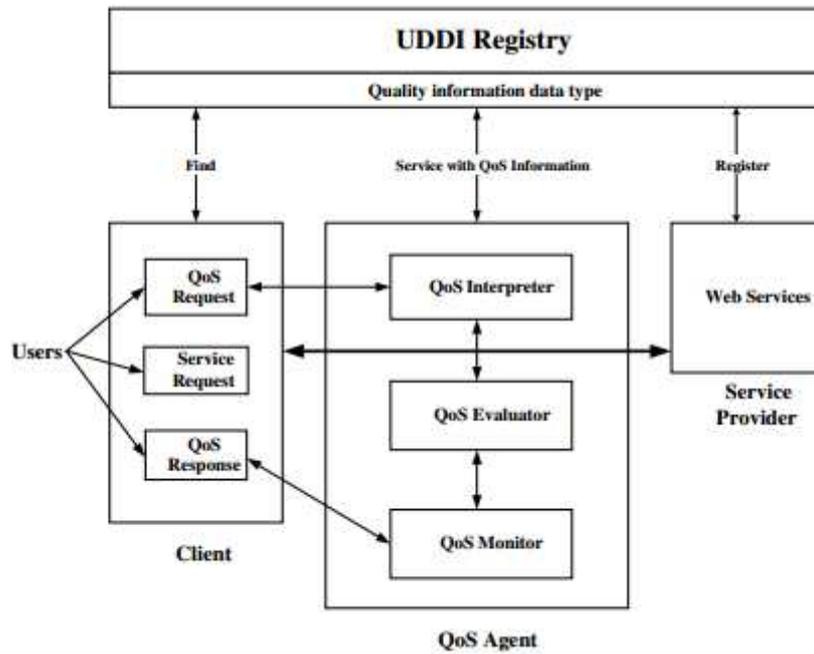
This approach contributes to extending the *WS architecture* to support QoS information manipulation. Besides, it combines proposals to specifying the QoS. In carrying out this function of specification, it uses semantics enriched WS- policies based on OWL ontologies and ARL rules, and QoS-enriched service publication and discovery using UDDI registries. These extensions are used by brokers for managing the QoS for WSs. Furthermore, in testing the effectiveness of this approach, the researchers developed a partial implementation of it and it was experimentally proven that the extensions could help users to select services in conformity with functional and non-functional requirements.

**Figure 17:** Web service architecture
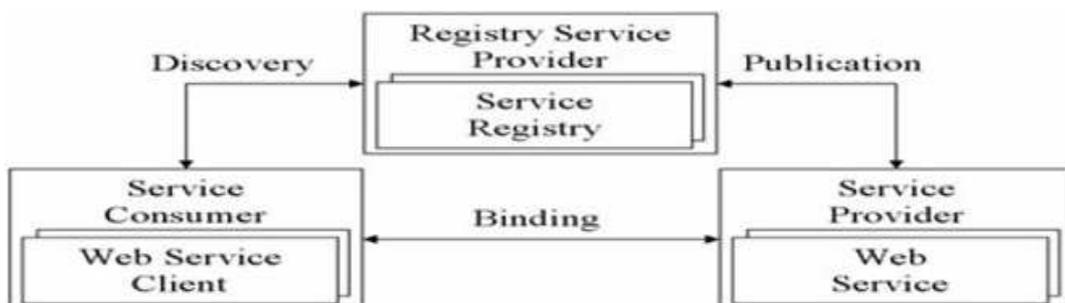
E. Al-Masri *et al* (2009) proposed a solution to overcome such a problem in this area and carried out a key study to gain further insights and understanding of service discovery goals. The researchers insert the concept of Quality of Web Service (QWS) in their quality-driven ranking mechanism. The study also provided a possibility of defining these service discovery goals as exploratory or informational. Furthermore, these findings were used to justify the great worth and the beneficial part of the knowledge of service discovery goals in enhancing the way how clients perform service search queries. Based on the results obtained from these experiments, the researchers found improvement in the performance of informational service queries in terms of precision in the querying process by 36.26% and it was estimated 40.39% in comparison to Google's Page Rank and Yahoo, respectively. Besides, these findings provided insights into developing the service retrieval process.

They provided a proposed extension of SOA based on autonomic registries. The latter model permits managing WSs and autonomic registries. The researchers also introduced a proposed rich WS information model based on the WS-Policy specification. This model includes both QoS descriptions and specific adaptation policies. Moreover, in this work, the researchers offered an extension of the original UDDI information model with new data structures for storing and managing the service related information.

Figure 18: Extended UDDI information model
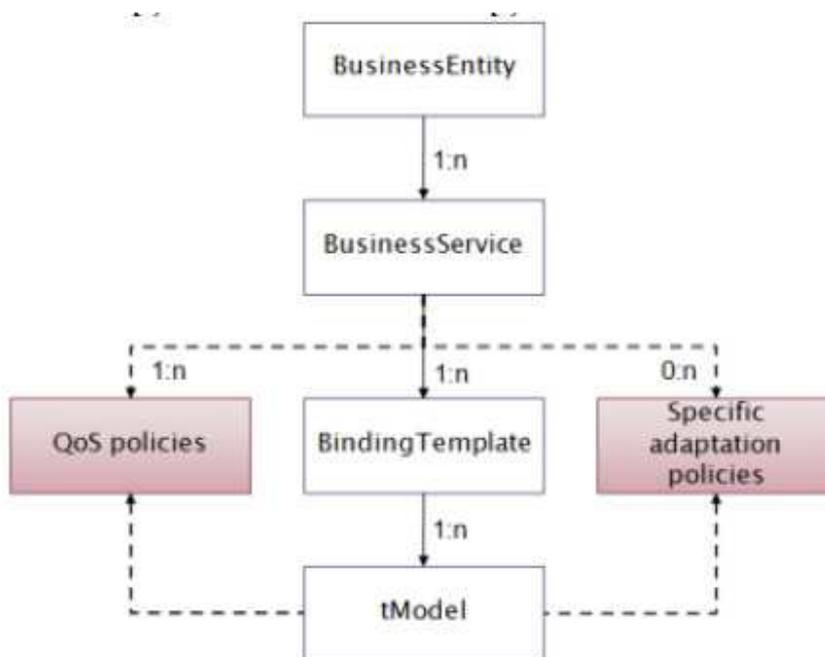
According to Cao *et al* (2009), the QoS-based WS evaluation is considered as a splendid promising research topic in dynamic WS selection. However, the major weakness of research in this area is ignorance of users' constraints and preference of the QoS while meeting individual users' requirement is so vital for selecting a satisfied WS. The researchers introduced an extended architecture capable of co-existing with the current WS publication and discovery model, and implemented a prototype to improve a better way to meet the functional and non-functional requirements of the user. Therefore, this preference model was introduced to Q-WSEM that enables the evaluating results to satisfy individual users' preference more effectively.

Figure 19: Extended Service Discovery Model
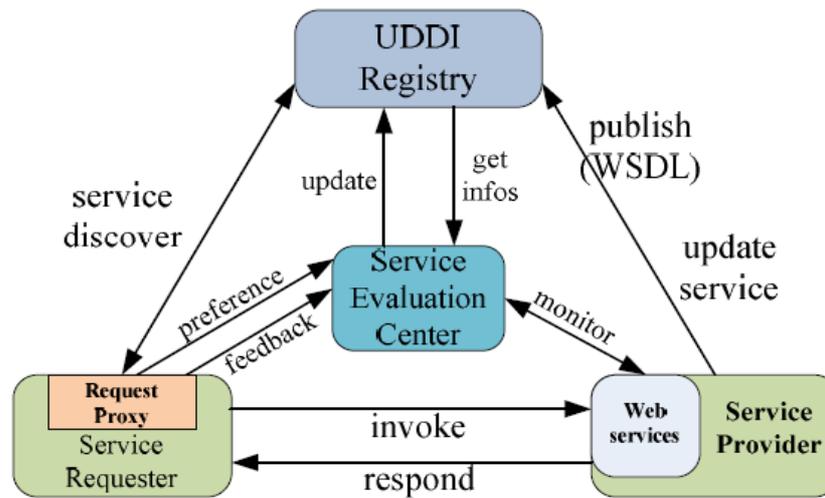
For WSs, Dancheng *et al*. (2010) introduced a new research on automatic generation of testing data. First, the researchers presented the development and related work of test data for WSs. Then, they defined the data structure of testing data, and introduced the algorithm of generation of testing data. Lastly, they validated the research via an experiment frame. WSs are more of a standard rather than a technology. Most of platforms offered diverse implements and development frameworks based on the different standards of WSs. In fact, this work contributed to current research in this area by combining the testing data and development framework, reducing the difficulty of the development of WS test and increasing the efficiency of Web test.

As pointed out by Al Hunaity *et al*. (2008), WS technology plays a significant major role in today's distributed system computing, service discovery. One of the main challenging difficulties that deter WS technology is identifying the available WSs that can be used by new web applications. However, the current solution which is based on UDDI catalogue browsing provides a support only for primitive matching mechanisms and provides no control over the quality of registered services. Thus, the major aim of this work was to refine the discovery process through setting up a new framework that enhances retrieval algorithms. This could be carried out by combining syntactic and semantic matching of service and also supporting the QoS information exploiting the environment knowledge form users' experience of service invocation and service initial run tests results.

Ming-Qiang Guo *et al*. (2009) suggested a distributed WS directory to afford servicing information for service requestors of SOA-oriented urban spatial information sharing platform. Thus, this paper contributes to previous research in terms of distributing the architecture and breaking the traditional mechanism of the UDDI registry by combining the portal WS directory and node WS directory. Moreover, this work eases the hardship of WS directory and enhances its efficiency and practicality. Another contribution is the possibility for various departments of setting up their own node WS directory and reasonably organizing private or semiprivate services registered library and easily implementing mutually communications among them. Thus, this prevents the implementation from bottlenecks and difficulties in maintaining information which are posed by the centralized model and it improves the SOA-based application services availabilities.

Conclusion

In consequence these services are nowadays accessible to the final clients. In the last few years, more and more Web Services providing the same functionalities are available in the environment. In order to select the best service adapted to client's requests, we need some method capable to evaluate and compare different services providing the same functionalities. In this context, Quality of service can be

defined as the capability to respond to the requirements (constraints) of a client and to fulfill these needs with the best criteria (preferences) established by the client. It is calculated based on the non-functional properties of the service. This paper provides an overview of a research progress in Quality of Service Based Web Services Discovery; it also highlights the issues that need to be investigated in Quality of Service Based Web Services

Acknowledgment

References

[1]   R. Dinesh Kumar et al (2011).” A QOS AWARE QUANTITATIVE WEB SERVICE SELECTION MODEL”, International Journal on Computer Science and Engineering (IJCSE) 2011.

[2]   Z. Xu, P. Martin, et al (2007).” Reputation-Enhanced QoS-based Web Services Discovery “2007 IEEE International Conference on Web Services (ICWS 2007)

[3]   K. Kritikos and D. Plexousakis, "OWL-Q for Semantic QoS-based Web Service Description and Discovery", SMRR, 2007.

[4]   Y.Baocai and Y. Huirong, et al (2007). “A Framework and QoS Based Web Services Discovery” 2010 IEEE

[5]   Lu Li and G. Zhang (2010), “A Management Framework for Web Service QoS based on Multi-Dimension QoS” The 5th International Conference on Computer Science & Education Hefei, China. August 24–27, 2010.

[6]   R.J. Retna Raj, Dr.T.Sasipraba (2011),” Web Service Recommendation Framework Using QoS Based Discovery and Ranking Process” IEEE-ICoAC 2011.

[7]   M. Alrifai, D. Skoutas , T. Risse (2010). “Selecting Skyline Services for QoS-based Web Service Composition”. 2010

[8]   V. X. TRAN (2008), “WS-QoSOnto: A QoS Ontology for Web Services” 2008 IEEE International Symposium on Service-Oriented System Engineering.

[9]   G. YE, C. WU, J. YUE, S. CHENG, C. WU (2009), “A QoS-aware Model for Web Services Discovery “2009 First International Workshop on Education Technology and Computer Science.

[10]   H. Yang,S. Liu,P. Fu,H. Qin,L. Gu (2009), “A Semantic Distance Measure for Matching Web Services “2009 IEEE

[11]   K. Kritikos and D. Plexousakis (2008), “Semantic QoS-based Web Service Discovery Algorithms for Over-Constrained Demands ” International Journal of Web Services Practices, Vol. 3, No.1-2 (2008), pp. 71-82.

[12]   Dr. I. Egambaram, G. Vadivelou, S. P. Sivasubramanian, “QOS BASED WEB SERVICE SELECTION ”,

[13]   K. Kritikos, D. Plexousakis (2007), “Requirements for QoS-based Web Service Description and Discovery”, International Computer Software and Applications Conference (COMPSAC 2007) IEEE.

[14]   L. Sha, G. Shaozhong, et al (2009), “A QoS based Web Service Selection Model” International Forum on Information Technology and Applications 2009 IEEE.

[15]   Q. Li-li, C. Yan (2009), “QoS Ontology Based Efficient Web Services Selection”, International Conference on Management Science & Engineering (16th) 2009.

[16] Chi-Chun Lo, Ding-Yuan Cheng et al (2008), "A Study on Representation of QoS in UDDI for Web Services Composition". International Conference on Complex, Intelligent and Software Intensive Systems. 2008 IEEE.

[17] E. Al-Masri and Q. H. Mahmoud (2007), "Discovering the Best Web Service", 2007, May 8–12, 2007, Banff, Alberta, Canada. ACM 978-1-59593-654-7/07/0005.

[18] C. Zhou and H. Chen (2009), "An Objective and Automatic Feedback Model for QoS Evaluation", 2009 ACM 978-1-60558-710-3/09/11.

[19] Z. Hui and G. Weiying (2009), "A Research on QoS-based Ontology Model for Web Services Discovery", 2009 IEEE DOI 10.1109/WKDD.2009.209.

[20] T. Rajendran and P. Balasubramanie (2009) , "An Efficient Framework for Agent-Based Quality Driven Web Services Discovery" 2009 IEEE.

[21] L.Taher , H. El Khatib , R.Basha , "A Framework and QoS Matchmaking Algorithm for Dynamic Web Services Selection " The Second International Conference on Innovations in Information Technology (IIT'05)

[22] Lou Yuan-sheng et al (2010), "A QoS-based Web Service Dynamic Composition Framework" 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science 2010 IEEE.

[23] N. Patil and A. Gopal (2010), "Ranking Web-services Based on QoS for Best-fit Search ", International Journal of Computer Science & Communication Vol. 1, No. 2, July-December 2010, pp. 345-349.

[24] A. Sedighi, "Modeling the Aspects of Quality of Service in Web Services and Service Oriented Architectures". Retrieved 22 august 2012 from http://www.ggf.org/mail_archive/graap-wg/2004/07/pdf00000.pdf

[25] N. Kokash "Web Service Discovery with Implicit QoS Filtering". Retrieved 22 august 2012 from http://homepages.cwi.nl/~kokash/documents/PhDICSOC.pdf.

[26] N. Patil and Dr. A. Gopal, "Enhancing UDDI registry for storing Qos in Model for discovering web services", Retrieved 22 august 2012 from http://www.sinhgad.edu/IJCSA-2012/pdfpapers11/11.pdf.

[27] R.Rajmohan , N.Padmapriya, S.K.V.Jayakumar (2011). "A Survey on Problems in Distributed UDDI", International Journal of Computer Applications, December 2011.

[28] Kee-Leong Tan et al (2007), "Personalization of SOA Registry Query Results: Implementation, Performance Analysis and Scalability Evaluation", World Academy of Science, Engineering and Technology 7 2007.

[29] T.Rajendran, Dr.P.Balasubramanie and R. Cherian (2010), "An Efficient WS-QoS Broker Based Architecture for Web Services Selection" 2010 International Journal of Computer Applications (0975 – 8887).

[30] P. Bonderud et al, "S-QoS4WS: Secure Quality of Service for Web Services Using the 3rd Party Certifications", Retrieved 22 august 2012 from http://www.tacoma.uw.edu/sites/default/files/global/documents/institute_tech/pbonderud.pdf.

[31] Q. Tao et al (2012). "A novel prediction approach for trustworthy QoS of web services "Expert Systems with Applications 39 (2012).

[32] C. Patel, K. Supekar, and Y. Lee. "A QoS Oriented Framework for Adaptive Management of Web Service based Workflows", Retrieved 22 august 2012 from http://www.uddi.org/pubs/Iru UDDI Technical White Paper.pdf.

[33] M. Rathore and U. Suman (2011). "A Quality of Service Broker Based Process Model for Dynamic Web Service Composition" Journal of Computer Science 7 (8): 1267-1274, 2011.

[34] N.Sasikaladevi and Dr.L.Arockiam (2010). "Reliability Evaluation Model for Composite web services "international journal of web & semantic technology 2010.

[35] V.X. Tran et al (2009) "A new QoS ontology and its QoS-based ranking algorithm for Web services", Simulation Modeling Practice and Theory (2009).

[36]   J.M. Ko et al (2008) "Quality-of-service oriented web service composition algorithm and planning architecture", The Journal of Systems and Software 2008.

[37]   M.B. Juric et al (2009), "WSDL and UDDI extensions for version support in web services", The Journal of Systems and Software, 2009.

[38]   V. Diamadopoulou et al (2008), "Techniques to support Web Service selection and consumption with QoS characteristics "Journal of Network and Computer Applications 2008.

[39]   M. Liu et al (2009). "An weighted ontology-based semantic similarity algorithm for web service " , Expert Systems with Applications 2009.

[40]   H.-C. Wang et al (2007), "Combining subjective and objective QoS factors for personalized web service selection", Expert Systems with Applications 2007.

[41]   D. Zuquim Guimaraes Garcia et al (2006). "Semantics-enriched QoS Policies for Web Service Interactions" ACM 2006.

[42]   E. Al-Masri et al (2009), "Understanding Web Service Discovery Goals "IEEE International Conference on Systems 2009.

[43]   H. Mezni et al (2011), "An Autonomic Registry-Based SOA Model "IEEE International Conference 2011.

[44]   J. Cao (2009), "QoS and Preference based Web Service Evaluation Approach "International Conference on Grid and Cooperative Computing 2009.

[45]   LI Dancheng et al, (2010), "The Research on Automatic Generation of Testing Data for Web Service" IEEE 2010.

[46]   M. Al Hunaity et al, (2008), "A New Web Service Discovery Model Based on QoS "IEEE CONFERENCE PUBLICATIONS 2008.

[47]   Ming-Qiang Guo et al (2009) "Design and Implementation of a Distributed Web Service Directory in SOA-oriented Urban Spatial Information Sharing Platform" Conference on Computational Intelligence and Industrial Applications. IEEE 2009.