# Taxonomy of Web Service Selection Approaches

M.Swarnamugi
Assistant Professor
Department of MCA
Jyoti Nivas Post Graduate Centre
Bangalore

## ABSTRACT
Web services are the prominent and newest technology for web application development that converge upon services as their basic element and permits application from different vendors to communicate with each other. The services along with XML protocols enables a number of service providers to provide loosely coupled and interoperable services at different quality of service and cost levels. Therefore the selection of an appropriate Web services has become a complex job due to tremendous growth of Web services offering similar functionality. It is vital to provide service consumers with facilities for selecting required web services according to their non-functional characteristics or quality of service (QoS). Therefore, the process of service selection is complicated due to divergent view of service consumers and service providers on the quality of services. The objective of this paper presents the exploration of various methodologies for web service selection process. A number of different approaches for methodologies have also been identified and presented.

## Keywords
Service Selection, Service Composition, Web Semantics.

## 1. INTRODUCTION
Service-Oriented Computing (SOC) is a new computing paradigm that uses services as fundamental elements to support the development of application processes in a rapid, low cost and heterogeneous environment. Services are autonomous entities that can be used in a platform independent way. Services can be described, published, discovered, selected and composed for building distributed application in an interoperable way. Services are built in a manner that is independent of the context in which they are used to develop an application. This means that the service provider and the consumers are loosely coupled. Key to this concept is the service-oriented architecture (SOA).

The Service Oriented Architecture (SOA) is "an architecture that represents software functionality as discoverable services on the network" [1]. Web Services are the dominant implementation platform for SOA, it uses a set of standards, SOAP, UDDI, WSDL, which enable a flexible way for applications to interact with each other over networks. SOAP is the new standard for network communication between software services. It is a general-purpose technology for sending messages between endpoints. The easiest way to publish a software component as a web service is to use a SOAP container. Once a component has been published as a web service, any SOAP-enabled client that knows the network address of the service and the messages that it understands can send a SOAP request and get back a SOAP response. To get the address and message information, SOAP clients read a WSDL file that describes the web service. Fortunately, most SOAP containers will automatically generate WSDL for the web services that they host, so developers don't have to write WSDL manually unless they really want to. Once the WSDL file is read, the client can start sending SOAP messages to the web service. WSDL describes what a web service can do, where it resides, and how to invoke it. UDDI is a new standard that allows information about businesses and services to be electronically published and queried. Published information is stored into one or more UDDI registries, which can be accessed through SOAP.

All these standards are XML-based (Extensible Markup Language), which allows applications to interact with each other over networks, no matter what languages and platforms they are using. The two features, self-description and language-platform-independence, distinguish web services from other distributed computing technologies, like CORBA (Common Object Request Broker Architecture) and DCOM (Distributed Component Object Model).

Web service have many challenging research issues, the most important among them is web service composition. Web service composition addresses the situation when a client's complex request cannot be satisfied by any single service, but by combining "parts of" available services it ca be achieved. Composition involves three different issues [2]. The first, called selection of service is concerned with selecting suitable services to composite that satisfy the user requirement an needs. The second, called composition synthesis is concerned with synthesizing a specification of how to coordinate the component services to fulfill the client request. The third issue, called as orchestration is concerned with how to actually achieve the coordination by executing the specification produced by the composition synthesis and by suitably supervising and monitoring both the control flow and the data flow among the involved services.

In this paper we present a complete study of service selection approaches for service composition. The paper is organized as follows. Section II describes the overview of service selection methodologies. In Section III, we discuss the approaches for identified methodologies. Finally section VI concludes with discussion and we highlight new challenges need to be addressed.

## 2. OVERVIEW OF SERVICE SELECTION METHODOLOGIES
The current Web services architecture and semantic Web addresses the problem of service discovery but not of service selection. Discovery deals with finding service implementations that meet a specified description. In the same way, selection deals with choosing a service implementation from among those that are discovered for the given description. Discovery is a prerequisite for selection, but it is selection which is the main problem. In order to use Web services successfully, both aspects must be addressed. Service description is handled by DAML-S and other competing standards such as UDDI and WSDL. The DAML-S authors acknowledge the importance of automatic service selection.

For this purpose, DAML-S includes various functional attributes, which include some quality of service parameters. For any service selection approaches the basic requirements include: Customer service requirement, Service offerings by the service provider and aggregating the evaluation results.

## 2.1 Customer Service Requirement

The customer service requirement may be simple or complex. Simple requirement may not look for composite services to satisfy the user query. Whereas, the complex requirements may have both functional and non-functional aspects, which needs to be satisfied. For this kind of complex requirement, the services need to be composite.

## 2.2 Service Offerings

The services offered by service provider also concerns about functional and non-functional aspects. The functional properties make use of domain ontology. To provide consumer the requested service with non-functional properties makes use of QOS ontology. The problem that arises here is how to map the quality preferences offered by consumer with the quality categorization in QOS ontology. This can be solved by labeling the qualities (eg. performance, security) in QOS ontology with the user preferences. This is called Service Categorization.

## 2.3 The Evaluation Results

This involves matching the customer required service with the offered services. This takes into account the representation of service execution path.

In basic form, service selection involves mapping a set of services to a service request—this can be thought of as the best service; in a more general form, service selections maps a set of services to a ranking of the services in that set. Multitude of service selection techniques and algorithms are proposed in the literature. Such as Use of optimization algorithm [3] for service selection, integer linear programming [4], broker-based architecture etc [5]. Yet the taxonomy of various approaches of service selection for service composition is not standardized.

We studied a number of criteria of web service selection process and with the use of these criteria we identified the following methodologies.

1. Functional based service selection methodology
2. Non-Functional based service selection methodology
3. User based service selection methodology

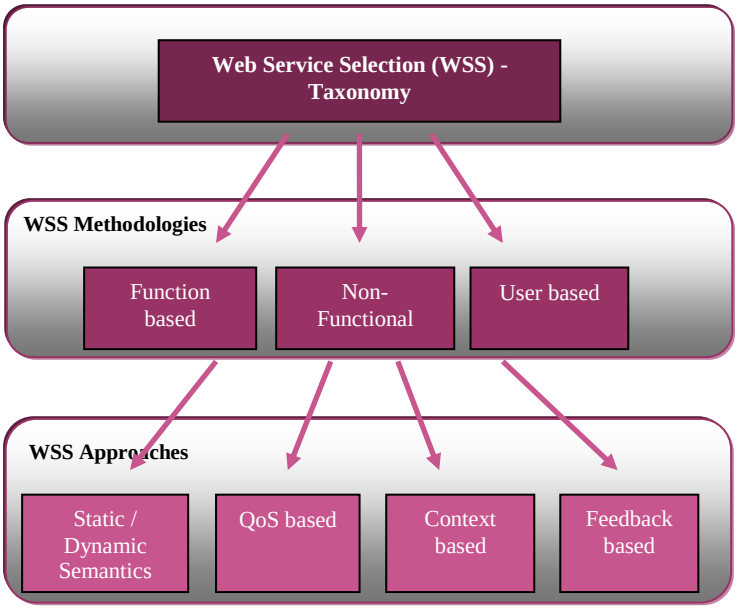


**Figure 1: Taxonomy of Web Service Selection Approaches**

Figure 1 illustrates the taxonomy of Web service selection. We broadly classify the web service selection as Functional based methodology, Non-functional based methodology and User based methodology. Functional based service selection methodology represents the Static and Dynamic semantics based approach. Selecting the appropriate service is often a question of retrieving functional descriptions from service repositories and then ensuring that the described and required interfaces match. Static semantics represents the properties of messages and operation semantics. The properties of messages include parameter passed (Data type, language, unit and business role) and message types (Serviceability, provider type, purpose, consumer type). Dynamic semantics represents the properties of behavior and operation logic.

When dynamic semantic is used in the selection process of Web service, the result of the selection contains more than one provider. With the rapidly growing number of available services, customers are presented with a choice of functionally similar services. This choice allows customer to select services that match other criteria, often referred to as non-functional attributes. Two fundamental questions arise because of this: How can these extra attributes be described and how can we select the most appropriate service. These questions should address both the selection of isolated services as well as the selection of services within the context of other services. The non-functional based service selection represents the QoS and Context in semantic web service selection. The properties of QoS may be (security, reliability, response time, call cost etc.), the properties of Context may include context of customer (location, intention, consumer's name, application, e-mail, termination of hardware and software) and context of service (provider's details, service descriptions etc,). User based represents the selection of best service among numerous discovered services based on customers' feedback, trust and reputation.

# 3. WEB SERVICE SELECTION APPROACHES

## 3.1 Methodology I: Functional based Service Selection

Realization of the full potential of the Web services requires technological advances in the areas of service interoperation, discovery, selection, composition and orchestration. A possible solution to all these problems can be provided by converting Web services to Semantic Web. Semantic Web services (SWS) can constitute a solution to the integration problem, as they enable dynamic, scalable and reusable cooperation between different systems and organizations. In general, the semantics to be added to a Web service may be called as functional semantics. In Web services, functional semantic is taken into consideration thereby avoiding unsatisfied results which are not of customer interest. Functional property is the functional semantics of a service that describes what a service actually does. The service functionality of a service is represented by a pair of its action and the object of the action. A hybrid semantic Web service selection of semantic services in SAWSDL based on logic based matching as well as text retrieval strategies are proposed. The first version of hybrid SAWSDL Web service matchmaker called SAWSDL-MX is proposed [6]. It exploits both crisp logic-based matching (subsumption reasoning) and IR-based (text retrieval) matching.

**(1) Static and Dynamic semantics based approach**

Web Service Selection is related to the process of evaluating and ranking the discovered web services to identify the ones that fulfill a set of functional and non-functional properties requested by the service customer. Most of the existing techniques rely on syntactic descriptions of service interfaces to find web services with disregard to semantic service parameters. This generates major problems in the service selection mechanism. To solve some of problems, Web service descriptions are enhanced with annotations of ontological concepts, semantic matching and by considering non-functional properties.

The various techniques used for service selection based on functional semantics in the literature have its own advantages with their earlier works done. It may not be possible by the techniques to fulfill all the specification we listed for functional semantic based service selection. That is one technique may be rich in one aspect of specification and the other on another aspect.

***Advantages*** – One advantage of this approach is eliminating the irrelevant services when compared with traditional syntactic keyword based service selection.

***Challenges*** - There may exist Web services that provide similar functional semantic properties, which might lead to the problem of differentiating available services

## 3.2 Methodology II : Non –Functianal based Service Selection

In a Web environment, multiple WSs may provide similar functionalities with different Non-functional property values (e.g., different prices). Such Web services will typically be grouped together in a single community. To differentiate the members of a community during service selection, their non-functional properties need to be considered. These properties are characterized as quality of service (QoS) and context based services. Both are highly important and are to be taken into account during the WS selection.

***(1) QoS based service selection approach***

The W3C working group (2003) defined various QoS attributes for web services (WS) in their 25th November 2003 publication. It comprises a number of generic items for cross-reference between the possible needs from service consumers and the functions supported by web services. This include: performance, reliability, scalability, capacity, robustness, exception handling, accuracy, integrity, accessibility, availability, interoperability, security, network-related QoS requirements etc. Although regular QoS attributes are listed, it remains some challenging problems on selection of web services. First, there exists some web services provided similar functional features, such as flight booking or restaurant reservation services, which might lead to the problem of differentiating available services with QoS. Further, perception on QoS attributes generally is different between consumer's preferences and providers. In addition, not all web services would like to expose the related QoS information for comparisons. A number of studies for overcoming these challenges have been carried out.

A QoS property can be static or dynamic. A static property has its value defined by a prior whilst a dynamic property requires measuring and updating its value periodically. The tendency attribute specifies the expected impact on QoS value from the service requester's perspective which can be positive, negative, close, or exact. For example, some QoS properties like price, response time have negative tendency (lower values are better) whereas throughput, availability have positive tendency (higher values are better). The importance level of QoS properties can be specified by defining its weight and mandatory. The weight attribute represents the priority level of a QoS property over the others whereas the mandatory attribute indicates that the satisfaction of a QoS property is strongly required or it is optional. The valid period of a QoS property can also be specified and updated periodically. The weight, mandatory, and valid period include not just the value but also the service participants which may define different values for the same QoS property.

QoS based service selection plays an important role in the process of service composition. QoS overcomes the problem faced in functional based service selection like they provide similar functional semantic properties, which might lead to the problem of differentiating available services. The approaches discussed above have advanced the process of QoS-aware service selection. QoS service selection overcomes the problem of similar service retrieval. However, they also fail to address the following issues:

- The issue of representation of QoS characteristics and QoS modeling.
- The issue of QoS weightings.
- The issue of different service customers' views on QoS characteristics and providers' fuzzy view on the QoS attributes.

***Challenges:*** With the proliferation of Web services as a business solution to enterprise application integration, the QoS offered by Web services are becoming the chief priority for service providers and their service consumers. Due to the dynamic and unpredictable nature of the web, providing the suitable QoS is really a challenging task. In addition to this, the different applications that are collaborating for Web services interaction with different requirements will compete

with each other. This makes service providers to understand about QoS characteristics. Also, a better QoS for a Web service will bring competing advantage over others by being a unique selling point for the service provider. The measurement process for each QoS metric is very complex since it should consider what and how to measure, who does the measuring and where the measurements are taken. This raises the issue of conflicts on QoS characteristics metrics between service consumer and provider.

***(2) Context based service selection approach***

Selecting the right parties to interact with is a fundamental problem in open and dynamic environments. The problem is exemplified when the number of interacting parties is high and the parties' reasons for selecting others vary. To achieve solution for this, services can be selected based on the context information. Context focuses on consumer perspectives and also in service perspectives. The main aim of Context aware service selection is to achieve the effectiveness. Context is used to facilitate the development and deployment of context-aware and adaptable Web services. Standard Web services descriptions can then be enriched with context information and new frameworks to support this enrichment can be developed. Context-based service selection for Web services composition has gained more advantages.

Context contains information about the execution environment of an activity that supplies information in application payloads. Management of the basic context type is facilitated by services defined in this specification. The specification also provides service interfaces for managing session-oriented protocols and representing the corresponding activities with contexts. The overall architecture of the context is hierarchical and decomposable, e.g., it is possible to use the context structure without reference to any activity model. In the field of Web services, context is used to facilitate the development and deployment of context-aware and adaptable Web services. Standard Web services descriptions can then be enriched with context information and new frameworks to support this enrichment can be developed.

The first element of the WS-Context specification is the context structure. The context structure defines a normal model for organizing context information. It supports nesting structures (parent-child relationships) for related contexts, and mechanisms to pass context information by reference or by value. A single context type is not sufficient for all applications; it must be extensible in a manner specific to a referencing specification and Web services must be able to augment the context, as they require.

In the field of Web services, context is used to facilitate the development and deployment of context-aware and adaptable Web services. Standard Web services descriptions can then be enriched with context information and new frameworks to support this enrichment can be developed. Context-based service selection for Web services composition has gained more advantages.

## 3.3 Methodology III: User based Service Selection

Used based methodology has been recognized as playing an important role in decision making in the Internet World. Customers and sellers must trust themselves and the services they are offered. Trust refers to the subjective probability by which an individual A expects that another individual B performs a given action on which its welfare depends. This definition taken from sociology is very popular in computer science today. From the business point of view, the European Commission Joint Research Centre defines trust as the property of a business relationship, such that reliance can be placed on the business partners and the business transactions developed with them. Reputation is what is generally said or believed about a persons or things character or standing. They argue that reputation is a mean of building trust, as one can trust another based on a good reputation. Therefore, reputation is a measure of trustworthiness, in the sense of reliability.

**(1) Feedback based service selection approach**

A User based methodology is a mechanism using consumers' feedbacks to identify good services from bad ones. Compared with other methodologies, it has more advantages in solving the selection problem for Web services. Trust and reputation play an important role in a service selection process of used based methodology. It is natural that a service consumer would like to choose a service that is trusted or a service with a high reputation. That is why trust and reputation mechanisms are used for making a good selection. With this approach, web service selection may be customized according to users' different constrains and preferences. Most approaches proposed in the literature about personalized selection concentrate on how to rank web services according to users' preferences on various QoS metrics. A trust based methodology [7] for service selection is proposed. QoS-based semantic web service selection solution with the application of a trust and reputation management method is presented. This work is based on Virtual Internet Service Provider.

User based methodology is a mechanism using consumers' feedbacks to identify good services from bad ones. Compared with other approaches, it has more advantages in solving the selection problem for Web services. Trust and reputation play an important role in a service selection process. It is natural that a service consumer would like to choose a service that is trusted or a service with a high reputation. That is why trust and reputation mechanisms are used for making a good selection. With this approach, web service selection may be customized according to users' different constrains and preferences.

# 4. DISCUSSION AND CONCLUSION

In this paper we have outlined three methodologies of Web service selection taxonomy that we feel, based on requirements of several case studies and thorough consideration of the topic, that allows users to select services based on functional, nonfunctional and used based properties.

It can be concluded that most approaches contribute specific aspects to the overall picture of service selection, which requires methods for expressing user requirements, expressing service offerings and also the actual service selection method. Approaches tend to concentrate on specific of these areas and employ a variety of techniques to do that. We feel that it is more appropriate to make some suggestions for future developments in the area of selection approaches. One interesting aspect of the work is that, it addresses very specific issues and tries to make contributions in these. There does not seem to be any work trying to address the overall spectrum of service selection and full set of requirements.

Some more aspects that need addressing are powerful mechanisms to capture user requirements that are both user friendly and also expressive enough to capture large numbers

of preferences and the logical relations between preferences. One aspect that falls into this area is the measuring of weights. Also, in the process of capturing the needs of users, their preference, specification of data, research has to show interest and capability to automatically capture this, to reduce the burden on the user part, and to react to changes in circumstances automatically.

## 5. REFERENCES

[1] Thomas Erl, "Service-Oriented Architecture (SOA): Concepts, Technology, and Design", Edition: 2005.

[2] Roy Grønmo, Michael C. Jaeger, "Model-Driven Methodology for Building QoS-Optimised Web Service Compositions." In Proceedings of the 5th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS'05), pp: 68–82, Athens, Greece, May 2005. Springer Press.

[3] Wentao Zhang Yan Yang Shengqun Tang Lina Fang Wuhan Univ, Wuhan, "QoS-driven Service Selection Optimization Model and Algorithms for Composite Web Services ", Proceedings of IEEE International Conference on Computer Software and Applications, pp: 425 – 431, 2007

[4] Danilo Ardagna, Barbara Pernici, "Global and Local QoS Guarantee in Web Service Selection", Springer LNCS, pp: 32 – 46, 2006.

[5] M.Adel Serhani, Rachida Dssouli, Abdelhakim Hafid, Houari Sahraoui, "A QoS Broker Based Architecture for Efficient Web Services Selection", Proceedings of IEEE International Conference on Web Services, pp: 113 – 120, Washington, US, 2005.

[6] Matthias Klusch, Patrick Kapahnke, "Semantic Web Service Selection with SAWSDL-MX", German Research Center for Artificial Intelligence, 2008.

[7] Galizia, S. Gugliotta, A. Domingue, J. Open Univ., Milton Keynes, "A Trust Based Methodology for Web Service Selection", Proceedings of IEEE International Conference on Semantic Computing, pp: 193 – 200, CA, 2007.