

# Enhancing Web Service Selection by User Preferences of Non-Functional Features

Youakim Badr, Ajith Abraham, Frédérique Biennier and Crina Grosan\*

*National Institute of Applied Sciences of Lyon, INSA-Lyon, F-69621 Villeurbanne, France*

*\*Department of Computer Science, Babes-Bolyai University, Kogalniceanu 1*

*Cluj-Napoca, 3400, Romania*

*youakim.badr@insa-lyon.fr, ajith.abraham@ieee.org, frederique.biennier@insa-lyon.fr,  
cgrosan@cs.ubbcluj.ro*

## Abstract

Selection of an appropriate Web service for a particular task has become a difficult challenge due to the increasing number of Web services offering similar functionalities. The functional properties describe what the service can do and the non-functional properties depict how the service can do it. Non-functional properties involving qualitative or quantitative features have become essential criteria to enhance the selection process of services making the selection process more complicated. Chaari et al. [8] proposed an ontological framework for modeling and exploiting non-functional properties. In this paper, we propose a simple Web services selection scheme based on user's requirement of the various non-functional properties and interaction with the system. The proposed framework utilizes user preferences as an additional input to the selection engine and the system ranks the available services based on the requirement. The method is validated using a popular Web services bench mark and the experiment results indicate that the scheme is useful and warrants further research.

## 1. Introduction

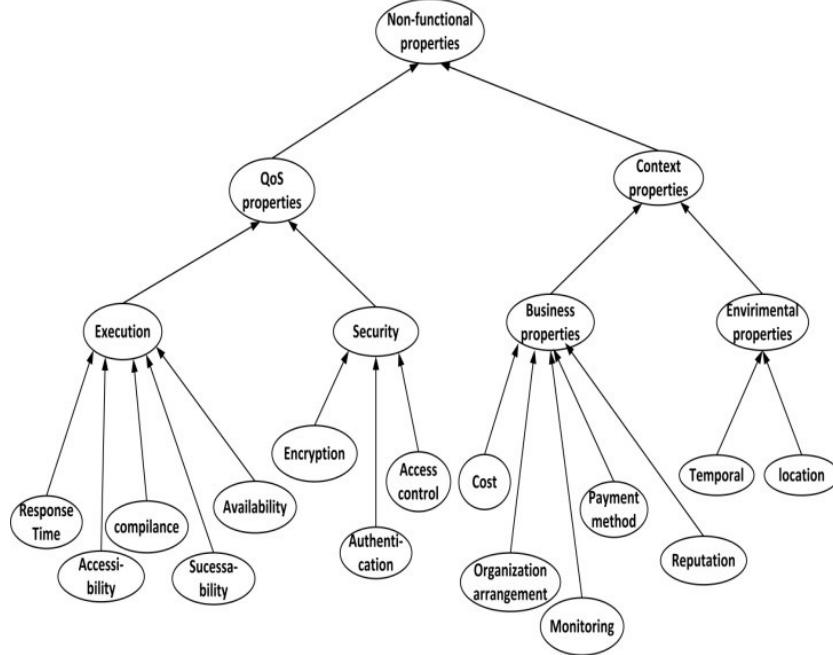
Web services are autonomous software systems identified by URIs which can be advertised, located, and accessed through messages encoded according to XML-based standards and transmitted using Internet protocols. Considering Web services, the basic standards are SOAP, WSDL, and UDDI, together they address the fundamental principles of the Service Oriented Architecture (SOA): publish-find-bind. Currently, this process is largely based on a human user making the decisions as to which service is appropriate for their purpose. Matching is mostly

based on functional requirements while non-functional properties are not formally considered. The increasing availability of Web services that offer similar functionalities with different characteristics increases the need for more sophisticated discovery and selection processes to match user requests.

Discovery corresponds to the activity of locating a Web service that meets certain functional criteria. Selection is related to the activity of evaluating and ranking the discovered Web services to identify the ones that fulfill a set of non-functional properties requested by the actual user. Most of the existing techniques rely on syntactic descriptions of service interfaces to find web services with disregard to non-functional service parameters. Previous research demonstrates how this situation generates major problems [1][2].

To solve some of problems, Web service descriptions are enhanced with annotations of ontological concepts, semantic matching and by considering non-functional properties (NFPs) [3][4]. We attempt in this present work to enhance web service discovery and selection involving user interaction by considering all the available functional and non-functional properties. As managing NFP raises the need for a structured representation of NF attributes, we consider treating them as policies of Web services. Since WS-Policy allows only for syntactic description of service properties, the NFP policies may be extended with additional semantic information using ontologies.

The rest of the paper is organized as follows. Section illustrates the NFP aware Web services model followed by the proposed User interaction based selection scheme in Section 3. Experiment illustrations are given in Section 4 followed by some conclusions.



**Figure 1.** NFP Categorization Ontology

## 2. A NFP-aware Web Services Model

A Web service may be fully described by two sets of properties: (i) functional and (ii) non-functional properties. One or more non-functional properties can be associated to a Web service and each property may be further divided into either QoS-related or context-related (see Figure 1).

**QoS Properties Category:** QoS-related properties represent a very important aspect of non-functional characteristics for a Web service. QoS may be further divided into two main categories:

**Execution:** includes the performance parameters which characterize the interaction with the Web service. We consider the following 5 features [1]:

1. *Response Time*: time elapsed from the submission of a request to the time the response is received.
2. *Accessibility*: represents the degree that a Web service is able to serve a request.
3. *Compliance*: represents the extent to which a WSDL document follows WSDL specification
4. *Successability*: represents the number of request messages that have been responded.
5. *Availability*: represents the percentage of time that a service is operating.

**Security:** is related to the ability of a given Web service to provide suitable security mechanisms by considering the following three parameters.

1. *Encryption*: the ability of a Web service to support the encryption of messages.
2. *Authentication*: the capacity of a Web service to offer suitable mechanisms dealing with the identification of the invoking party and allow operation invocation.
3. *Access control*: whether the Web service provides access control facilities to restrict the invocation of operation and the access to information to authorized parties.

**Business Properties Category:** Like QoS properties, they are relevant for differentiating Web services having the same functional characteristics and we consider two main categories of business properties:

**Strategic properties:** include the following five features:

1. *Cost*: represents money that a consumer of a Web service must pay in order to use the Web service.
2. *Reputation*: measures the reputation of Web services based on user feedback.
3. *Organization arrangement*: includes preferences and history (ongoing partnerships)
4. *Payment method*: represents the payment methods accepted by a Web service, i.e. transfer bank, Visa card etc.

5. *Monitoring*: required for a number of purposes, including performance tuning, status checking, debugging and troubleshooting.

**Environmental properties** include two features namely *location* and *temporal properties*.

Modeling the non-functional properties refers to the language to be used for representing the non-functional properties of Web services. As managing NFP raises the need for a structured representation of such properties, we consider them as policies of Web services.

### 3. Weighing User Preferences

The proposed method considers all the available non-functional properties of the available Web services but gives more importance to the most preferred properties and the remaining properties are given equal importance. To model this, the weighted sum approach is used which is a traditional, popular method that parametrically changes the weights among objective functions to obtain the Pareto front [5].

Let us consider we have the objective functions  $f_1, f_2, \dots, f_n$ . This method takes each objective function and multiplies it by a fraction of one, the "weighting coefficient", which is represented by  $w_i$ . The modified functions are then added together to obtain a single cost function, which can easily be solved using any method which can be applied for single objective optimization.

Mathematically, the new function is written as:

$$\sum_{i=1}^n w_i \cdot f_i$$

where  $0 \leq w_i \leq 1$

$$\text{and } \sum_{i=1}^n w_i = 1$$

The initial work on the weighted sum method was done around 1963 by Zadeh [6].

Trade-off surface may be generated by varying the weights  $w_1, \dots, w_n$ . The main weakness of weighting technique is the method allows only linear relationships among the criteria. Particularly the method fails in the presence of non-convex search spaces. The approach can not generate all Pareto optimal solutions for non-convex trade-off surfaces. The method is simple to understand and easy to implement. The weight itself reflects the relative importance (preference) among the objective functions under consideration.

Instead of specifying the real value for the non-functional properties, the user has to simply specify the importance of the different attributes, which makes this scheme easy to use even for someone not very familiar with the different attribute values. If the user considers all properties are important then the weights are distributed equally. If user considers only certain attributes are important then the weights will be distributed equally between the other remaining attributes.

### 4. Experiment Results

In this section, some implementation aspects will be discussed in order to evaluate the feasibility of the proposed approach.

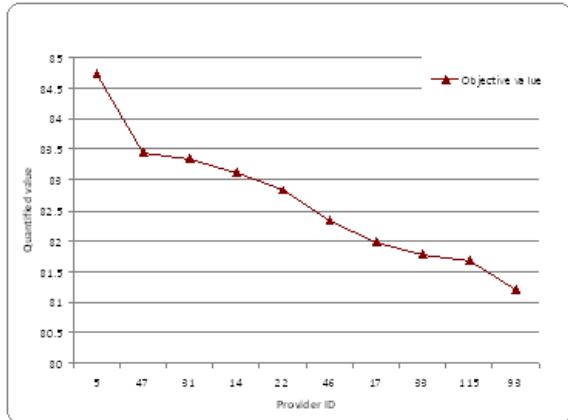
We used a Web service database which includes 364 Web services [7]. The database contains the name of the services and its non-functional parameters values. Each Web service is represented by 14 non-functional property features: *response time*, *availability*, *throughput*, *successability*, *reliability*, *compliance*, *best practices*, *latency*, *documentation*, *cost*, *reputation*, *payment method*, *security* and *monitoring*.

#### Case I

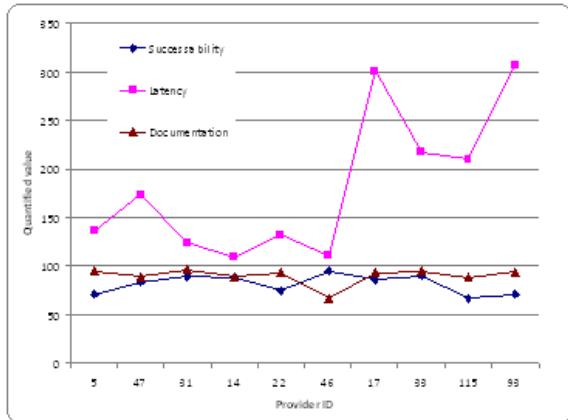
In the first illustration, we consider the user option: All features are important. The weight for each feature in this case = 1/14. Figures 2-5, illustrate the weighted sum approach for the top 10 service providers. Even though service provider ID 5 has the maximum objective value, it has relatively high cost and low successability compared to others. Provider ID 93 has the lowest cost, while provider ID 5 has the highest reputation. Please note that the entire top 10 service provider's differ marginally in terms of objective value (Figure 2). Once the user is able to visualize the top service providers, then a suitable decision shall be made based on further preferences (example: lowest cost, etc.).

#### Case II

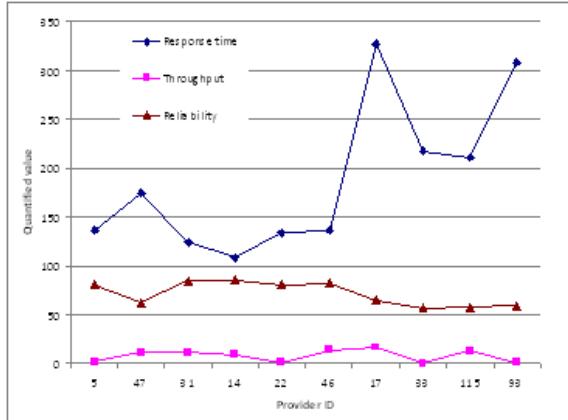
In the second illustration, we consider the user preference as follows: 50% importance for cost, 20% importance for reliability and 20% for reputation. In this case, weights for the three features specified are 0.5, 0.2 and 0.2 and the weights for the remaining 11 features are 0.1/11. Figure 6, illustrates the objective values for the top 10 providers using the weighted sum approach. Figures 7-9 illustrate other performance matrices. As evident from Figure 6, provider 351 has the modest cost, with the highest reputation and reasonable reliability.



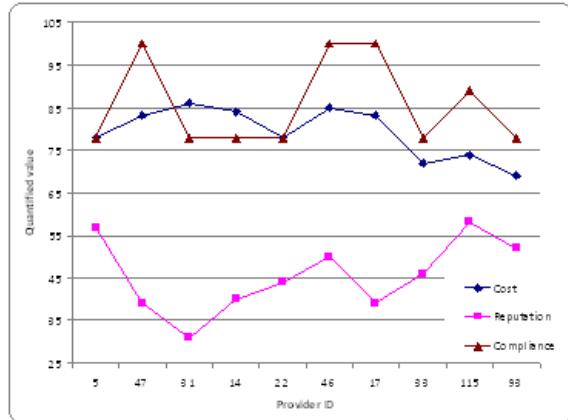
**Figure 2.** Objective Value of Top 10 Services (Case 1)



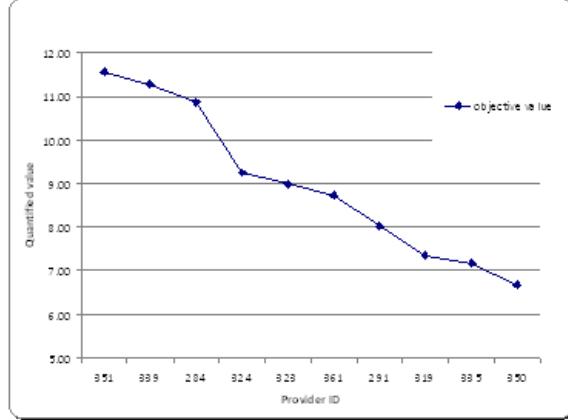
**Figure 3.** Top 10 Services Showing Successability, Latency and Documentation (Case 1)



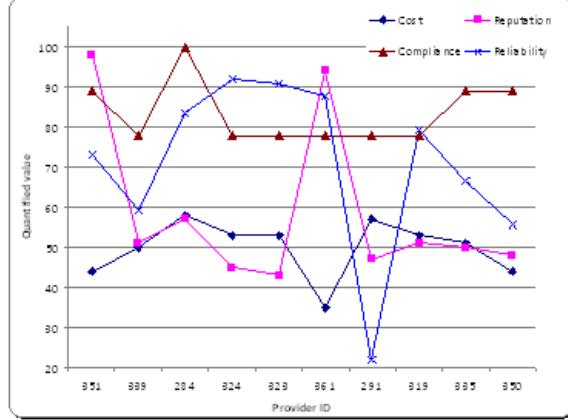
**Figure 4.** Top 10 Services Showing Response, Throughput and Reliability (case 1)



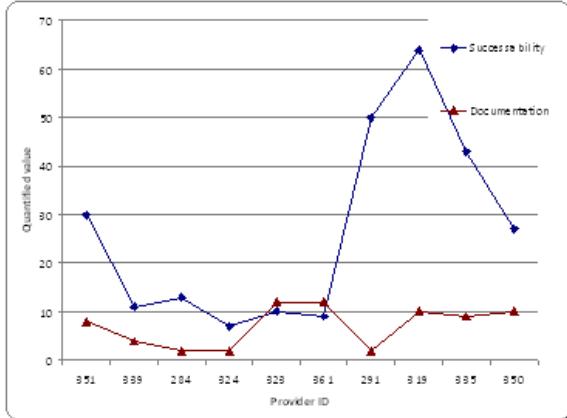
**Figure 5.** Top 10 Services Showing Cost, Reputation and Compliance (Case 1)



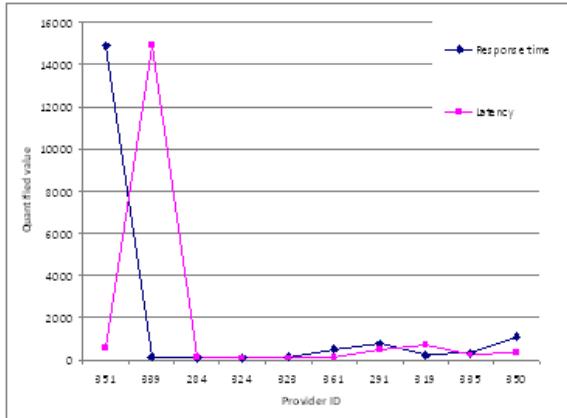
**Figure 6.** Objective Value of Top 10 Services (Case 2)



**Figure 7.** Top 10 Services Showing Cost, Reputation, Compliance and Reliability (Case 2)



**Figure 8.** Top 10 Services Showing Successability and Documentation (Case 2)



**Figure 9.** Top 10 Services Showing Response Time and Latency (Case 2)

## 5. Related work

Non-functional properties exhibit constraint over the functionality provided by web services. They increase the semantic richness of service descriptions and provide necessary pre-requisites for automated discovery and selection. Different taxonomies attempt to classify these properties. O'Sullivan et al. [15] propose a classification taxonomy including availability (temporal and locative), payment, price, obligations, rights, security, trust, quality, discounts, and penalties. Other works [16, 17] address taxonomies which vary in complexity and detailedness. However, non-functional properties remain crucial for web service discovery and selection. Several research works deal with functional properties to select services and limit non-functional properties [18] [19]. The first attempt to tackle this approach considers the most essential ingredient of service representation by capturing the non-functional properties of services [2]. The authors argue that accurate service descriptions of non-

functional parameters support essential activities such as discovery, composition, substitution and management of services. Nevertheless, this work enumerates only some NFPs and does not focus on how users can influence the selection with their preferences. Several academics and industrial researches have addressed the integration of QoS in Web service standards[3-5] in order to enhance specification, publishing and discovery. In [6], the authors propose methods and ontology for service selection taking into account a set of non-functional parameters. The key difference between our contribution and previous works relies on the ranking model which takes into account user preferences and the fine-tuning of the selection accuracy to fit user expectations.

## 6. Conclusions

This paper proposed a simple Web services selection scheme based on user's requirement of the various non-functional properties and interaction with the system. The proposed framework utilizes user preferences as an additional input to the selection engine and the system ranks the available services based on the user preferences. The proposed architecture also relies on selection and matching engines, which interact with service communities.

**Acknowledgement:** This work received support from the French National Research Agency (Agence Nationale de Recherche) through the "SEMantiquE pour bUS de sErvice" (SemEUse) project under the funding grant Number 2007 TLOG 2008.

## References

- [1] K. Kritikos and D. Plexousakis, "Semantic QoS Metric Matching" in *European Conference on Web Services (ECOWS'06)*, 2006, pp. 265-274.
- [2] E. M. Maximilien and M. P. Singh, "Toward Autonomic Web Services Trust and Selection," in *International Conference On Service Oriented Computing 2004*, pp. 212-221.
- [3] S. A. Chun, V. Atluri, and N. R. Adam, "Using Semantics for Policy-Based Web Service Composition," *Distributed and Parallel Databases*, vol. 18, p. 37, 2005.
- [4] M. Comerio, F. D. Paoli, A. Maurino, and M. Palmonari, "NFP-Aware Semantic Web

- Services Selection," in *11th IEEE International Enterprise Distributed Object Computing Conference*, 2007, pp. 484-491.
- [5] Kim IY, de Weck OL (2005), Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and Multidisciplinary Optimization*, 29, 149-158.
- [6] Zadeh L (1963), Optimality and Non-Scalar-Valued Performance Criteria. *IEEE Transaction on Automation Control* 8, 59-60.
- [7] E. Al-Masri and Q. H. Mahmoud, "QoS-based Discovery and Ranking of Web Services," in *IEEE 16th International Conference on Computer Communications and Networks*, 2007, pp. 529-534.
- [8] S. Chaari, Y. Badr, F. Biennier, C. BenAmar, J. Favrel, Framework for Web Service Selection Based on Non-Functional Properties, *International Journal of Web Services Practices*, Volume 3, No. 2. pp. 94-109, 2008.
- [9] M. Ouzzani, "Efficient Delivery of Web Services," PHD thesis, vol. 15, 2004.
- [10] J. O'Sullivan, D. Edmond, and A. t. Hofstede, "What's in a Service? Towards Accurate Description of Non-Functional Service Properties," *Distributed and Parallel Databases*, vol. 12, pp. 117-133, 2002.
- [11] Y. Liu, A. H. H. Ngu, and L. Zeng, "QoS Computation and Policing in Dynamic Web Service Selection," in *World Wide Web*, 66-73, 2004.
- [12] D. A. Menasce, "QoS issues in web services," *IEEE Internet Computing*, vol. 6, pp. 72-75., 2002.
- [13] X. Wang, T. Vitvar, M. Kerrigan, and I. Toma2, "A QoS-aware Selection Model for Semantic Web Services," in *International Conference on Service Oriented Computing 2006*, pp. 390-401.
- [14] M. Comerio, F. D. Paoli, A. Maurino, and M. Palmonari, "NFP-aware Semantic Web Services Selection," in *11th IEEE International Enterprise Distributed Object Computing Conference*, 2007, pp. 484-491.
- [15] Justin O'Sullivan, David Edmond and Arthur HM ter Hofstede. Formal description of non-functional service properties. Technical Report FIT-TR-2005-01, Centre for Information Technology Innovation, Queensland University of Technology, 2005.
- [16] Suzanne Robertson and James Robertson. Mastering the requirements process. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1999.
- [17] Donald G. Firesmith. Common concepts underlying safety, security, and survivability engineering. Technical Note CMU/SEI-2003-TN-033, Carnegie Mellon Software Engineering Institute, 2003.
- [18] John Garofalakis and Yannis Panagis and Evangelos Sakkopoulos, Web Service Discovery mechanisms: looking for a needle in a haystack, *International Workshop on Web Engineering 2004*
- [19] Shuping Ran, A Model for Web Services Discovery with QoS, *ACM SIGecom Exchanges*, vol 4, pp 1-10, 2003