A Holistic Method for Selecting Web Services in Design of Composite Applications

Mārtiņš Bonders, Jānis Grabis

Institute of Information Technology, Riga Technical University,
1 Kalku Street, Riga, LV – 1658, Latvia, martins@iti.rtu.lv

Abstract. Selection of appropriate web services is an important step in development of composite applications. Quality of Service (QoS) data characterizing nonfunctional properties of candidate web services are usually used in web service selection. Functional characteristics, which are difficult to measure, and composite application's lifecycle management factors are equally important. The paper develops a comprehensive approach for selecting web services in design of composite applications. This approach accounts for factors related to non-functional and functional requirements and to composite application's lifecycle management. The service selection is performed using a generic mathematical optimization model. Establishing relationships between composite application's lifecycle management processes and web service selection is the key challenge in elaboration of the selection model.

Keywords. Web services, QoS, ITIL, Service Lifecycle Management

1 Introduction

Composite applications use external services to gain access to vast data and processing resources. Web services relying on such open technologies as XML, SOAP, WSDL, REST and others are often used as building blocks for composite applications. That allows developing standards-based, scalable and flexible applications. Properties of composite applications directly depend upon characteristics of external services used and environmental factors, which in the case of public networks, exhibit high degree of variability. Therefore, selection of appropriate and reliable services is of major importance. Multiple methods have been elaborate for selection of such services from the set of candidate services providing similar functionality [1,2,3]. However, majority of these methods focus on non-functional attributes of web services as a main factors influencing web service selection. [4] shows that functional aspects have an equally significant impact. Additionally, the composite application incorporating the selected services should be supported throughout its lifecycle, and the selected services also have impact on lifecycle management and costs of composite applications.

The objective of this paper is to develop a comprehensive web service selection model for design of composite applications. The model should account for both non-functional and functional requirements as well as for lifecycle management factors of

the composite application. The web service selection problem is formulated as an optimization problem. This optimization problem is an inherently multi-objective problem. In order to account for this characteristic, impact of service selection is measured in monetary terms.

The paper presents the general holistic approach to web services selection and develops a generic optimization model for service selection. These developments will serve as a basis for future research on elaboration of the comprehensive optimization model and on development of a platform for prototyping and experimenting with composite applications.

Section 2 describes the general approach for web service selection. A generic mathematical programming model for service selection is given in Section 3. Section 4 outlines future research activities.

2 Approach

Web service election methods often use Quality-of-Service (QoS) measurements to evaluate services. These measurements including reliability, security, trust and execution cost are mainly concerned with nonfunctional characteristics of services. However, service consumers are equally concerned about both functional and nonfunctional characteristics of services and successfully management of developed composite application. There have been attempts to expand the QoS concept in the case of web service selection by defining it as: ‘‘The degree to which a system, component or process meets customer or user needs or expectations” [1]. This definition includes evaluation of both functional and nonfunctional requirements. Unfortunately, formal evaluation of functional characteristics in the framework of web service selection is more difficult than evaluation of nonfunctional characteristics. Functional requirements are represented either in a simplified form or require judgmental appraisal of degree of their satisfaction. From the perspective of managing composite applications throughout their lifecycle, the selected web services have direct impact on development and maintenance. This issue has been largely ignored in service selection literature.

The propose web service selection approach attempts to account for both non-functional and functional requirements as well as for composite application’s management aspect (Fig. 1). The non-functional requirements are accounted for using QoS measurements such as response time and reliability. The functional requirements are accounted for using quantitative performance measures of the composite application. Finding quantitative performance measures characterizing functional requirements is difficult. One approach could be measuring application’s impact on business key performance indicators though these indicators give only indirect measurements, they are affected by different side-effects and can be obtained only after the application has been deployed and used for some time. Decision-making applications form one class of applications for which quantitative performance measurements can be determined because functionally these applications are required to produce specific performance indicators. For instance, if composite application is used in vehicle routing then the main functional requirement is finding a route

between two locations. In this case, alternative services can be evaluated by comparing travel times returned by the services.

From the composite application's lifecycle management perspective, the composite application itself is perceived a service, which requires appropriate maintenance and support. Standards like Information Technology Library (ITIL) [5] govern management of such services throughout their lifecycle. They include a number of management processes and not all of them are affected by service selection. Processes affected by web service selection include change management (external services should be continuously monitored for changes in service definition and performance) and information technology architecture management (impact of using different web service integration methods). Some other processes directly depend upon QoS characteristics of web services, for instance, availability management.

The web service selection model incorporates all factors mentioned above and is used to select services to be used in development of the composite application. A generic form of the service selection model is presented in the following section.

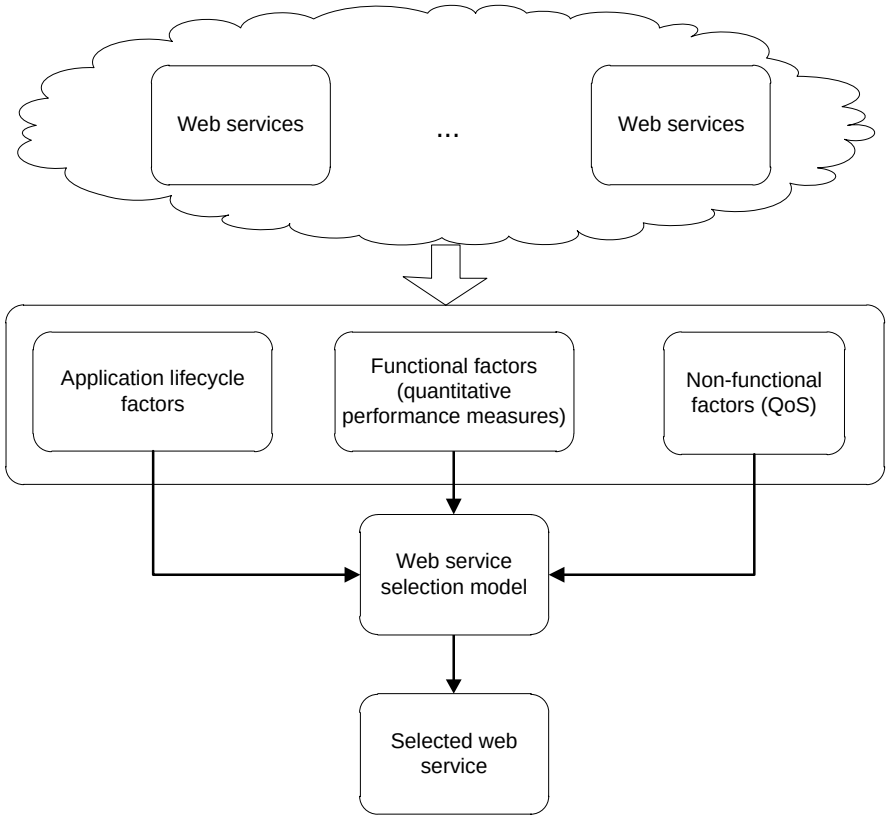

Fig. 1. The web service selection method

3 Mathematical Model

It is assumed that requirements for a composite application relevant to web service selection are given by N functional requirements referenced by index i and M nonfunctional requirements referenced by index j. Additionally, there are R parameters referenced by index r characterizing lifecycle properties of the composite application depending upon services used in its design. There are K alternative web services $\mathbf{S} = (s_1, ..., s_K)$ satisfying the requirements to some degree. QoS data $\mathbf{G}_k = (g_{k1}, ..., g_{kM})$ characterize how well the kth service performs according to the jth nonfunctional requirement. Quantitative performance measures $\mathbf{H}_k = (h_{k1}, ..., h_{kN})$ characterize how well the kth service performs according to the ith functional requirement. Lifecycle properties $\mathbf{L}_k = (l_{k1}, ..., l_{kR})$ characterize impact of the kth service on rth property of the composite application. A web service selection model takes $\mathbf{G}_k$, $\mathbf{H}_k$ and $\mathbf{L}_k$ as input data and finds the best service using anappropriate selection algorithm.

A mathematical programming is used as the selection method. In order to combine different functional and nonfunctional criteria, a universal optimization metric is used. This metric characterizes the cost of building and operating the composite application depending upon web services used in its development. It is computed as

$$TC = \sum_{k=1}^{k} \left(\sum_{i=1}^{N} c_i^f h_{ki} + \sum_{j=1}^{M} c_j^{nf} g_{kj} + \sum_{j=1}^{M} c_j^{lfc} l_{kj} \right) X_k \to \min ,$$

where X_l indicates whether the kth service is selected and c_i^f , c_i^{nf} and c_i^{lfc} are parameters characterizing costs associated with functional requirements, nonfunctional requirements and lifecycle management properties, respectively. The optimization is performed subject to relevant constraints.

The mathematical model is given in its generic form and should be augmented with appropriate constraints. However, the main challenge is estimation of the model's parameters. $\mathbf{G}_k$ can be determined according to data provided by the service provider or using simulation studies. In order to evaluate $\mathbf{H}_k$, a prototype composite application is built. A variant of the prototype P_k is developed for each candidate web service. Values of $\mathbf{H}_k$ are determined by executing the corresponding prototype P_k. Values of $\mathbf{L}_k$ are computed by taking into account application lifecycle management key performance indicators used at the enterprise.

4 Future Work and Conclusion

The paper has proposed a general approach to selecting web services for design of composite applications. The key distinctive feature of this approach is combination of factors related to non-functional and functional requirements and to composite application's lifecycle management. The approach uses the mathematical

programming model for service selection. The main subject of further research is estimation of parameters of this model.

Accounting for functional requirements asks for development of prototype composite applications for each combination of services to be evaluated. That is feasible only if web services can be substituted with relative ease (i.e., changing services should not affect the core of the composite application). Additionally, that would require implementation of a specialized framework for experimenting with composite applications.

The proposed approach for estimating impact of functional requirements is restricted to decision-making applications, which have clear quantitative performance measures. Possibilities to formally quantify satisfaction of functional requirements for other types of applications also should be explored.

Current web service selection methods are able to deal with accounting for non-functional factors, and [4] demonstrates incorporation of functional factors in the web service selection model while estimation of lifecycle properties of composite applications subject to selected web services is the subject of further research. In general, relationships between web service selection and development and operation of composite applications should be identified and formally defined.

References

1. Huang Angus F.M., Ci-Wei Lan, Stephen J.H. Yang, An optimal QoS-based Web service selection scheme, Information Sciences: an International Journal archiveVolume 179 , Issue 19, Pages: 3309-3322, 2009.
2. Buhwan J., Hyunbo C., Choonghyun L., On the functional quality of service (FQoS) to discover and compose interoperable web servicesExpert Systems with Applications 36, 5411–5418, 2009.
3. Dimitrios T., Ioanna R., Efstathios S., QoS-aware service evaluation and selection, European Journal of Operational Research Volume 191, Issue 3, Pages 1101-1112, 2008.
4. Bonders M., Grabis J., Kampars J. Combining Functional and Nonfunctional Attributes for Cost Driven Web Service Selection Frontiers in Artificial Intelligence and Applications // Databases and Information Systems VI - Selected Papers from the Ninth International Baltic Conference: DB&IS 2010. Vol.224, Latvia, Riga, 5.-7. July, 227-239. 2010.
5. Official ITIL® Website , http://www.itil-officialsite.com/home/home.aspx