

User-side Testing of Web Services

Gerardo Canfora

RCOST – Research Centre on Software Technology, University of Sannio, Benevento, Italy
gerardo.canfora@unisannio.it

Abstract

Service Oriented Architectures, and particularly Web Services, are receiving a growing attention from research and industry. With Web Services, software is used and not owned and operation happens on machines that are out of the user control. Therefore, providing users with means to build confidence that a service delivers the desired function with the expected QoS becomes a key issue.

1. Introduction

Service Oriented Architectures represent a model in which small, loosely coupled pieces of functionality are published, consumed, and combined with other functions over a network. Web Services are an implementation of such architectural model over the Internet.

Interoperability within and across enterprises is certainly one of the main promises of Web Services. However, long-term interest derives from the possibility to realize the “software-as-a-service” concept, i.e. a clear separation of the possession and ownership of software (*software as a product*) from its use (*software as a service*) [3]. This has several positive implications, primarily enabling a shift of the software market from the current technology-driven, vendor-led model to a business-oriented, demand-driven model. It also allows for decoupling the speed of development and evolution of a software system from the speed of change of its code components [1].

Needless to say, there are many problems, both technical and human related, that call for further research. Testing of a service to gain confidence that it delivers the desired function with the expected QoS is certainly a key problem to be addressed, as a lack of trust will severely prevent web service mainstream adoption.

2. User-side testing of Web Services

Service testing can be viewed from different perspective: the perspective of the service provider that owns the service implementation and is responsible of its evolution; and the perspective of the user, which uses the services without knowing its implementation, relying on the service interface. In this talk, I’ll focus on the user perspective; references [2, 4] are a few notable examples of papers that address the service provider perspective.

2.1. Generating test cases for QoS testing

The user of a service stipulates a SLA with the provider, and the latter agrees to ensure proper levels of QoS. QoS testing of a service will have the aim to broken SLA constraints, because of:

- different paths followed in the service workflow, which produce different QoS values;
- when late binding is enabled, different concrete services are invoked thus potentially broking SLA constraints.

I’ll discuss the use of evolutionary testing, in particular Genetic Algorithms, to search for combinations of inputs and concretizations of services bound to the service-under-test, such that the SLA is violated and paths with critical values of reliability and availability are identified.

2.2. Regression testing

Regression testing is the process of re-testing a service after changes have been made to ensure that these changes have not adversely affected the delivered function. A key problem is that the service provider may not be aware of who is using the service and hence cannot notify them when changes to the interface or the implementation of the service happen. Therefore, the responsibility for regression testing is shifted to the service user. This highlights the importance for web services to be provided with a test suite, accessible from the UDDI registry or from the service WSDL. I’ll discuss the need for exporting service test cases in an open XML-based format and I’ll illustrate research work aimed at automatically generating XML representation of test cases from JUnit (<http://www.junit.org>), linking this XML representation to the WSDL description of a service, and using the latter to generate test drivers on the client side.

3. Concluding remarks

The research work discussed in this keynote is part of a long-term research effort being developed at RCOST with the aim of developing methods and tools for service oriented system development and evolution.

References

- [1] G. Canfora, *Software Evolution in the Era of Software Services*, International Workshop on Principles of Software Evolution, September 2004, Kyoto, Japan.
- [2] C. Fu, B. Ryder, A. Milanova, D. Wonnacott, *Testing of Java Web Services for Robustness*, International Symposium on Software Testing and Analysis, July 2004, Boston, Massachusetts.
- [3] M. Turner, D. Budgen, P. Brereton, *Turning Software into a Service*, IEEE Computer, October 2003.
- [4] W. Xu, J. Offutt, *Generating Test Cases for Web Services Using Data Perturbation*, Workshop on Testing, Analysis and Verification of Web Services, July 2004, Boston, Massachusetts.