Verification of Reconfiguration Mechanisms of Service-Oriented Architectures

László Gönczy

As the automated integration of heterogeneous software environments becomes widespread, there is a growing demand for resilient software architectures. The Service-Oriented Architecture is an emerging paradigm in this field, however, in its present form, it does not cover the fault-tolerant aspects and no verifiable system reconfiguration mechanisms are modeled.

As the configuration of the underlying services may often change, (e.g. consider a network of mobile services), the system must be able to react to effects of the changes of the environment by means of dynamic reconfiguration. On the other hand, certain properties (such as the availability of the minimal number of service instances of a given service type, the presence of a particular service or some quantitative requirements) must be guaranteed during the entire lifetime of the system. Therefore, the aim of my ongoing research is twofold:

First, the components of a typical Service Oriented Architecture (services, ports, messages, etc.), their non-functional properties (such as the guaranteed response time or the acknowledgement options), the reconfiguration mechanisms (e.g. searching for and invoking a new service if a call fails or the response time decreases below a certain limit), and the fault model (e.g. a service crashes or becomes overloaded) will be described in a high-level model while the underlying technology, for instance, SOAP messages or Grid technologies will remain hidden. The basis of the followed approach is discussed in [2].

Second, I’ll implement a system for the verification of reconfiguration algorithms. Based on the high-level, technology-independent system description, i) the state space of the model will be investigated by reachability analysis to generate the set of the possible succeeding system configurations and ii) the fulfillment of the requirements against these configurations (and the transient states) will be verified. During this analysis, a graph transformation tool with a model checking support can be used, like Groove [1]. The most important future research task is the investigation of the applicability model-based synthesis of Service-Oriented Architectures, or, more precisely, service configurations.


---

1 This work was partially supported by the SENSORIA European project (IST-3-016004).