Simulation and Formal Analysis of Workflow Models

Kovács Máté. Inf. kovmate@hotmail.com

Advisors:
László Gönczy, Department of Measurement and Information Systems,
gonczy@mit.bme.hu
Dr. Dániel Varró, Department of Measurement and Information Systems,
varro@mit.bme.hu

Today, major companies tend to store the data related to business on electronic storage devices. This makes it possible that more than one task can be carried out on he same work unit parallelly in time. The workflow defines the order of the activities that has to be carried out on a work unit. In practice the BPEL (Business Process Execution Language [1]) is used to implement such process in an executable way. This enables the loose coupling of software components which run on different platforms. These systems are extremely dependent on uncontrollable external programs, so the dependability, and non-functional properties of these systems need to be verified.

However, the workflow is a semiformal notation. It has to be extended in order to prove some of its properties with mathematical accuracy. There is a proposal in which the workflows are mapped into nondeterministic automata, and then the properties of these are examined [2]. However, this transformation needs some human decisions.

In my work I discuss a method, which allows us to carry out the formal analysis of a workflow implemented in BPEL without human interference. This method relies on graph-transformations [3], that are executed by the Viatra (Visual Automated Model Transformations) [5]. The Viatra is an extendable modular framework. I implemented a plug-in which is able to import a BPEL file, and create it's representation inside the VIATRA modelspace. The first step is to transform the BPEL model into a dataflow network model [4], which can be done in a deterministic way, due to the BPEL's properties. The second transformation maps the dataflow into a Promela (Process Meta Language) model. After extracting the Promela code it can opened with the SPIN (Simple Promela Interpreter) [6], which can formally verify, whether an arbitrary temporal logical expression is valid for the concerned model. This way the dynamic properties of a BPEL model can be formally checked. In my work I demonstrate the usage of this tool by a pilot example, and discuss the possibilities of further development.

References:

1. Business Process Execution Language for Web Services Version 1.1. 2003 ftp://www6.software.ibm.com/software/developer/library/ws-bpel.pdf
2. J. Koehler , G. Tirenni, S. Kumaran: From Business Process Model to Consistent Implementation: A Case for Formal Verification Methods , EDOC 2002, pages 96-106.
3. dr. Bartha Tamás, dr. Csertán György, Gyapay Szilvia, dr. Majzik István, dr. Pataricza András, dr. Varró Dániel: Formal Methods in Information Technology, Typotex kiadó, Budapest, 2004 pages:215-242.
4. A.J. Anderson, Data Flow Systems. In Multiple Processing:Systems Overview, ch. 10. pp. 441-488, Prentice Hall, UK, 1989.
5. VIATRA2 Eclipse GMT subproject, http://www.eclipse.org/gmt/
6. Spin Version 4: Language Reference, http://spinroot.com/spin/Man/promela.html