TOWARD AN INTEGRATION OF BPEL IN AGAPIA

GHEORGHE ŞTEFĂNESCU[1], MONICA TĂTĂRĂM[1]

Agapia, a recently proposed high-level structured programming language for interactive computation, can be seen as a coordination language and developed on top of usual imperative or functional programming languages. (WS)BPEL (a Web Services Business Process Execution Language) is an orchestration language and focuses on execution and control of modern business processes. This paper reports preliminary results on an attempt to integrate BPEL in Agapia.

1. Agapia – A Structured Interactive Programming Language

Interactive computation [11][2] is a step forward on program modularization allowing to describe and verify programs in an open environment. Recently, a model (consisting of interactive systems with registers and voices – *rv-systems*) and a core programming language (for developing *rv-programs*) based on register machines and a space-time duality transformation have been introduced, [20]. Structured programming techniques for rv-systems and a kernel programming language AGAPIA have been subsequently introduced in [6, 7], with a particular emphasis on developing a structural spatial programming discipline.

Structured process interaction greatly simplifies the construction and the analysis of interactive programs. For instance, method invocation in current OO-programming may produce unstructured interaction patterns, with free goto's (by method invocation) from a process to another and should be avoided. Compared with other interaction or coordination calculi (e.g., π-calculus [14], actor models [1], REO [2], Orc [15], etc.), the rv-systems approach paves the way towards a name-free calculus and facilitates the development of a modular

[1] Department of Computer Science, University of Bucharest, gheorghe@funinf.cs.unibuc.ro; tataram@fmi.unibuc.ro

[2] The term "interactive computation" often refers to interactive systems where one participant is human, dealing with development of powerful human-computer interfaces. In the approach presented in this paper, there are multiple interacting participants, each being either human or machine "computing component".