Journal of Universal Computer Science, vol. 9, no. 11 (2003), 1258-1260
submitted: 29/10/03, accepted: 3/11/03, appeared: 28/11/03 © J.UCS

Formal Specifications of Computer-Based Systems

J.UCS Special Issue

Václav Dvořák
(Brno University of Technology, Czech Republic
dvorak@fit.vutbr.cz)

Miroslav Sveda
(Brno University of Technology, Czech Republic
sveda@stud.fit.vutbr.cz)

Charles Rattray
University of Stirling, UK
cr@cs.stir.ac.uk)

Jerzy W. Rozenblit
University of Arizona, USA
Jerzy.Rozenblit@arizona.edu)

This special issue presents a collection of papers from the last Workshop on Formal Specifications of Computer-Based Systems (FS-CBS 2003) that took place in Huntsville, Alabama in April 2003. Similar workshops were previously organized by devoted people - Charles Rattray (UK), Jerzy Rozenblit (USA), and Miroslav Sveda (CZ) - in Edinburgh, Scotland (2000), Washington, D.C. (2001) and in Lund, Sweden (2002) as parts of the annual IEEE International conferences on Engineering of Computer-Based Systems (ECBS). The Huntsville workshop was attached to the 10th ECBS conference. Since computer-based systems (CBS) are ubiquitous and get ever more diversified and sophisticated, the need of formal methods and especially formal specifications becomes more acute and urgent than ever. Subsystems labeled as embedded, real-time, intelligent, adaptive, re-configurable, fuzzy, autonomic and the like (beside mechanical, hydraulic or pneumatic subsystems) can all be found within CBS. It is clear that finding a common general description of such heterogeneous systems, particularly the use of formal notations to describe assumptions, requirements and a design of CBS, is a great challenge that is worth of every effort. That is why FS-CBS Workshops came into being.

The Formal Specifications Working Group within the IEEE Computer Society (TC-ECBS) is a forum for researchers and practitioners from industry and academia. In the last workshop, as also in the previous ones, discussions focused on completed work as well as on work-in-progress related to FS CBS, on software, hardware and mixed hardware/software applications. Authors of presented contributions from Europe and North America extended their submissions to full papers and after the reviewing process the best 5 have been selected for this Special issue of the J.UCS.

This is already the second special J.UCS issue of this kind since the first was published in November 2000.

The paper "*Monitoring Temporal Logic Specifications Combined with Time Series*" by Doron Drusinski and Man-Tak Shing shows using an extended Temporal Logic with time Series (TLS) for run-time monitoring important properties of CBS such as stability, monotonicity, temporal average, min/max and sum values. The novel TLS extension is based on practical experience with in-process and remote monitoring tools provided by NASA engineers when verifying flight code. Verification of timing properties in rapid system prototyping can then be done by analysis of schedulability and satisfaction of timing constraints in TLS. The example of a fish farm control system at the end is a nice illustration of the described formal methods and tools.

The next paper targets the growing need for reliable systems such as airport control systems, banking systems or medical instruments that have to be correct by design in order not to fail ever. The goal of a paper entitled "*Automatically Generated CSP Specifications*" by Frantisek Scuglik and Miroslav Sveda is to make verification of system correctness easier by automatically creating the system's formal model. The paper reports experience and results with home-made prototype tools related to two techniques of automated CSP (Communicating Sequential Processes) support: using either behavioral diagrams based on UML Composite States diagrams or the direct translation of application source code into CSP by a compiler. The results seem promising and open up further possibilities how to extend the tools in order to create more precise CSP specifications.

Model-based system development requires frequent transformations of models between design stages or tools. These transformations must be formally specified in order to maintain end-to-end semantic interoperability. The paper "*On the Use of Graph Transformation in the Formal Specification of Model Interpreters*" by Gabor Karsai, Aditya Agrawal, Feng Shi, and Jonathan Sprinkle addresses this problem by suggesting a graph-transformation-based technique for specifying these model transformations. If steps in construction of CBS are formally specified, then the correctness of a design can be verified via correctness of the steps. The paper defines the transformation language, its implementation and illustrates its use on an example. Thus the technique can be the first step on the road to achievement of correctness-by-construction property of CBS.

The paper "*Defining a Formal Coalgebraic Semantics for the Rosetta Specification Language*" by Cindy Kong, Perry Alexander and Catherine Menon explains how to use the dual of algebra, i.e. co-algebra, to formally describe semantics of specifications. Units of specification are facets and the system level design language titled Rosetta makes use of them for specification of state-based systems; the behavior of non-state-based systems can also be formalized even though not so directly. Interactions between facets are defined by means of functions between facets and domains and commuting diagrams are used to define these functions. We will see in future whether Rosetta proves itself in practical use and in what application segment. No doubt, a single modeling language that would satisfy the requirements of all CBS is probably still miles away.

The last paper "*An Information Flow Method to Detect Denial of Service Vulnerabilities*" by Stéphane Lafrance and John Mullins brings us to the very

up-to-date area of validation of security protocols. Whereas the validation of confidentiality and authentication policies has been studied extensively, denial of service (DoS) only sporadically, leaving some space for intruders´ attacks. Authors use extended Security Process Algebra (SPPA) and a cost-based framework in formal characterization of DoS. Comparing the cost to the defender and to the attacker is important in creation of more DoS-resistant protocols. Combining the cost-based framework with SPPA has led to new formal characterization of DoS (the concept of impassivity). Highly theoretical analysis is nicely complemented by application to the well-known TCP/IP protocol.

In conclusion we want to thank to all selected authors of this special issue for their effort in writing their high quality manuscripts. We hope that readers will benefit from the most recent work in formal specifications and that they will get interested in formal methods generally, and in formal specifications of CBS specifically. Enjoyable reading!

Czech Republic, UK and USA
November 2003

Václav Dvořák
Miroslav Sveda
Charles Rattray
Jerzy Rozenblit