Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 176 (2007) 1–4

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Preface

The aim of the FESCA workshop is to bring together researchers from academia and industry interested in formal modelling approaches, and associated analysis and reasoning techniques with practical benefits for embedded software and component-based software architectures.

Component-based software design has received considerable attention in industry and academia since object-oriented software development approaches became popular. Recent years have seen the emergence of formal and informal techniques for the specification and implementation of component-based software architectures. There are several specific reasons for the use of software component technology in embedded systems. Firstly, an embedded hardware system is usually build up from components, and it is natural to want to mirror the component-based hardware structure closely with a component-based software structure. Secondly, for various technical systems, product-lines exist. One way to support product-lines on the software side is to use component orientation, as this eases the configuration of software. With the expansion of application areas of embedded software-systems, this trend has been amplified.

One traditional means for developing safety critical applications and demonstrating their safety is through the use of formal methods. However, formal methods often do not scale well and cannot be adequately used for increasingly complex systems. Concurrency, mobility, distribution, component-orientation, various communication styles, and so on, are currently mainly covered by research approaches, while formal methods used in industry generally pose rather hard constraints on the system to be modelled and analysed. Often, engineers use semi-formal notations such as UML2 to model and organise components into architectures. FESCA aims to address the open problem of how formal methods can be applied effectively to these new contexts and challenges. FESCA is therefore interested in formal methods known from the area of embedded software development and software engineering, and tries to cross-fertilise their research and application.

One strength of FESCA lies in the link established between the embedded software design community and the formal software engineering community by exploring how formal approaches developed in one community affect or can be exploited by the other. Previous FESCA workshops achieved this by looking at new computing

1571-0661
doi:10.1016/j.entcs.2006.11.020

paradigms like ubiquity, component-orientation and novel middleware technologies, which are of shared interest for both embedded software design and formal software engineering.

Consequently, FESCA 2006 specifically called on other areas of shared interest to embedded systems and component-based software engineering:

Dependability: Proving the dependability of a system is of increasing relevance for most systems including embedded systems, mission-critical enterprise systems, e-Commerce, and so on.

Quality attributes and Resource Consumption: Quality attributes are of increasing interest for both domains. While current software development processes used in industry are mainly driven by the correct implementation of functional requirements, the systematic evaluation and prediction of quality attributes such as reliability, availability, resource consumption, performance and scalability are a matter of research. Given the complexity of today's concurrent, distributed and networked software, it is extremely important to provide formal techniques and CASE tools for the analysis and reasoning of local component properties and global system properties.

Consequently, FESCA is interested in exploring how formal analysis and prediction techniques widely used a specific domain can be applied in new contexts.

The success of FESCA over the years is notorious through the high quality of the submissions attracted. Common topics of the eight accepted papers presented include component models and specification, component selection and model checking. In addition to paper presentations, FESCA 2006 had as an invited speaker Frantisek Plasil who presented joint work with Petr Hnetynka on hierarchical and flat component models.

Throughout the workshop, the participants engaged in various discussions concerning general questions and open problems. As in previous workshops, there was a clear cross-fertilisation of members from the component-based software engineering and the embedded systems communities. In particular, the following areas were identified as very promising for future interactions between the communities.

Component Models: Much of the research and insights gained in CBSE concerns the definition of elaborated component models (in fact, component *meta* models). For example, the role of explicit context modelling when predicting quality attributes, the various dependencies of a component (and particularly its quality attributes) and its environment, the need of including information on the dependencies between provided and required interfaces of a component in its specification. Many of these insights are not tied to enterprise applications, and are expected to be useful in component models for embedded systems. Despite the success of the embedded systems community in transferring formal methods to commercial software development, there is usually a large delay in making use of advances in software engineering, for example, using high-level languages, object-oriented or component-based approaches.

Modelling of quality attributes with stochastic distribution functions:

To express real-time guarantees in the form of upper bounds is important in embedded systems for reasons of application safety. By contrast, the middleware of communication systems in enterprise systems does not provide any kind of real-time guarantees. As a consequence, research in quality of service of enterprise systems investigates stochastic distribution functions to specify and predict quality of service. The timing behaviour of many new embedded applications (such as mobiles, PDAs, etc) is not bound to safety properties of the systems as it traditionally was, and the use of stochastic models for quality attributes seems attractive also in this domain.

Model Checking: While research in protocol-modelling interfaces has a long history in CBSE, the use of model checkers for proving component interoperability has not yet been exploited. The inclusion of timing properties in interfaces and the use of recent work on timed model checking is a very promising research area for CBSE.

Given the usual high-quality of submissions received by FESCA, the organisers decided to introduce the FESCA best paper award to honour the contribution of the author(s) of the best paper. This year we had two outstanding papers and gave a shared best paper award to:

Björn Metzler and Heike Wehrheim (University of Paderborn, Germany) for their paper *Extending a Component Specification Language with Time*. All three reviewers suggested strong acceptance. To quote from the reviews: "The paper describes the solution to an important problem, the question how time constraints for components can be expressed and checked for concrete implementations. It is a smart decision of the authors to not create yet another formalism but instead find means to express time in existing formalisms. This offers not only the advantage that specifications are more readable. Also existing theory (the semantics of the specifications) and tools (model checker) can be extended and used for the specification variant with time, as has been demonstrated in the paper." and "In a strife for extending their well-researched CSP-OZ formalism to specification and analysis of real-time components, the authors pursue a a technically straightforward extension of CSP-OZ by introduction of clock variables, as found in timed automata. In contrast to other extensions of the same formalism aiming at the same domain, in particular CSP-OZ-DC (extending the formalism with Duration Calculus formulae, thus integrating a full-fledged metric-time temporal logic), the current extension is technically much simpler and also considerably less expressive. This is, however, well justified by the ease of manipulation gained: there is a direct embedding of timed CSP-OZ into (potentially infinite) timed transition table, which facilitates automatic model checking with standard tools if the state space of the specification happens to be finite."

Samik Basu (Iowa State University, USA), Partha S Roop and Roopak Sinha (University of Auckland, New Zealand) for their paper *Local Module Checking for CTL Specifications*. Again, the three reviewers all opted for strong acceptance: "This paper gives a local approach to CTL module checking by determining a single environment for which the negation of the property is satisfied. So the state-space of the module is explored locally and on-the-fly, so this technique only explores those states needed to prove the negation of the property." and "The paper introduces a new local method for module checking based on a tableau construction. Module checking is concerned with model checking open systems, where the environment may influence the holding of a temporal logic property (here CTL). The technique tries to construct an environment which - in combination with the system - falsifies the property under consideration." and "The main contribution of the paper is a sound and complete set of tableau rules for local CTL module checking. Though having the same worst case complexity as global module checking, the authors demonstrate that a practical implementation of their local approach yields much better results."

This volume includes the eight accepted papers at FESCA 2006. The continuous success of FESCA in attracting high-quality submissions and valuable discussions motivates the organisers to continue with FESCA in 2007 at ETAPS 2007 in Braga, Portugal.

Juliana K.F. Bowles
Iman Poernomo
Ralf Reussner
25 November 2006