

COMPONENT-BASED SYSTEM DEVELOPMENT EDITORIAL

Hartmut Ehrig* and Julia Padberg**

*Dept. of Computer Science
Technical University Berlin, Germany
ehrig@cs.tu-berlin.de
**FB 3, Universität Bremen, Germany
padberg@informatik.uni-bremen.de

This special issue is based on selected contributions at IDPT 2002, the 6th World Conference on Integrated Design and Process Technology, Pasadena, June 2002. The topic of component-based system development was one of the main topics at IDPT 2002, especially in the Distinguished Lecture Series on Software Components for Continuous Engineering.

The first contribution in this special issue on "Consistent Interaction of Software Components" by Engels, Küster and Groenwegen presents a methodology for consistency management for different UML-models including an application to components modeled in UML. In the second contribution on "Specification of Network Embedded Software Components", short NEST components, by Sünbül a framework for developing light-weighted, formally correct software is presented which is well-suited for NEST applications. A practitioner's view on component-based software development is given by Kriegel et al, which discusses a platform-independent markup-language for components together with supporting tools. The objective handicaps and drawbacks of component-based software engineering in a commercial setting are discussed in the last contribution by Löwe based on experience with the top five German insurance companies.

The remaining three contributions are concerning net based modeling techniques. The new concept of open Petri nets which is especially useful to model reactive systems, is applied by Heckel and Chouikha to the control synthesis for discrete event systems. The transformation-based component framework presented by Ehrig, Orejas et al. is a generic approach how to model components for different graph and net-based modeling techniques in a unified way. For the specific case of Petri nets an interesting module concept is presented by Padberg, where the core concept can be considered as an instantiation of the generic approach in the previous contribution.

In summary the various contributions in this special issue show some important aspects of component-based system development from the theoretical and the practical point of view. We hope that the reader can benefit from both views in order to improve component-based system development in the future.