Softw Syst Model (2005) 4: 233 / Digital Object Identifier (DOI) 10.1007/s10270-005-0096-z

Editorial

Modeling Paradigms

Robert France, Bernhard Rumpe

Published online: 17 June 2005 –

This issue contains three papers in its regular section and includes a special section that consists of extended versions of the best papers from the "St.Eve" (State versus Event-based Modeling) Workshop. In the state-based modeling paradigm, behavior is conceptualized and described in terms of state changes. Behavior in data intensive applications (e.g., business systems) and in control intensive systems (e.g., embedded controllers) fit well in this paradigm. Event-based modeling is particularly suited to describing systems in terms of interactions across their constituent interfaces in the early development phases (e.g., in requirements and high-level architecture phases). Focusing on interactions across interfaces in the early phases is good practice. It allows a developer to abstract out irrelevant internal details while gaining an early understanding of required interactions and constraints on how parts interact in an application. The focus on understanding interactions across interfaces in the early stages can also lead to early convergence of stable application architectures.

State and event-based modeling paradigms are not the only paradigms that are used when modeling software based systems and their context. For example, neither approach is well-suited to modeling workflows. In addition, physical or logical distribution and deployment, and threads of activity cannot be adequately described using events or states. It should not come as a surprise that the development of an application may require the use of multiple modeling paradigms. Integrating multiple modeling paradigms is one of the great challenges of model-drive development. An effective integration must be based on a deep understanding of the relationships among modeling paradigms. Workshops such as "St.Eve" help the community develop such an understanding.

Contents in this issue

This issue contains a special section that consists of extended best papers from the "**St.Eve Workshop**". This section is edited by *Tommaso Bolognesi* and *John Derrick*. Following this editorial, there is an introduction to the special section that gives an overview of the papers in this section.

The second part of this issue contains three regular papers. The regular paper "**Specifying business rules in object-oriented analysis**" by *Frank Devos* and *Eric Steegmans* introduces a language for describing business rules based on the OCL. The OCL is extended to support definition of framing rules and a mechanism to describe the effect of events is introduced. Class and event constraints are combined and provide yet another view on the combination of events and states.

In the regular paper "**Precise visual modeling: a case study**" *John Howse* and *Steve Schuman* describe their results on a case study that describes a "video rental service" using a visual formalism. The formalism is used to describe structure, pre- and postconditions and constraints in a graphical manner. The case study demonstrates that precision and expressiveness can be supported in visual modeling languages.

The third regular paper "**Testing Web applications by modeling with FSMs**" by *Anneliese A. Andrews, Jeff Offutt* and *Roger T. Alexander* describes an approach to effectively testing web-based applications using finite state machines (FSMs). They combine test case generation with a constraint-based mechanism that provides guidance to selecting appropriate test cases.

We hope you enjoy reading the articles in this issue.

Robert France, Bernhard Rumpe
Editors in Chief