Sixth International Workshop on Specification and Verification of Component-Based Systems (SAVCBS 2007)

ESEC/FSE 2007
6th Joint Meeting of the European Conference on Software Engineering and the ACM SIGSOFT Symposium on the Foundations of Software Engineering
Dubrovnik, Croatia
September 3-4, 2007

SAVCBS 2007
PROCEEDINGS

Specification and Verification of Component-Based Systems

http://www.eecs.ucf.edu/SAVCBS/

September 3-4, 2007
Dubrovnik, Croatia

Workshop at ESEC/FSE 2007
6th Joint Meeting of the
European Conference on Software Engineering and the
ACM SIGSOFT Symposium on the
Foundations of Software Engineering

Sixth International Workshop on Specification and Verification of Component-Based Systems (SAVCBS 2007), September 3-4, 2007, Dubrovnik, Croatia.

SAVCBS 2007
TABLE OF CONTENTS

SAVCBS 2007
ORGANIZING COMMITTEE

Mike Barnett (Microsoft Research, USA)
Mike Barnett is a Research Software Design Engineer in the Foundations of Software Engineering group at Microsoft Research. His research interests include software specification and verification, especially the interplay of static and dynamic verification. He received his Ph.D. in computer science from the University of Texas at Austin in 1992.

Dimitra Giannakopoulou (RIACS/NASA Ames Research Center, USA)
Dimitra Giannakopoulou is a RIACS research scientist at the NASA Ames Research Center. Her research focuses on scalable specification and verification techniques for NASA systems. In particular, she is interested in incremental and compositional model checking based on software components and architectures. She received her Ph.D. in 1999 from the Imperial College, University of London.

Gary T. Leavens (School of EECS, University of Central Florida, USA)
Gary T. Leavens is a professor in the School of Electrical Engineering and Computer Science at the University of Central Florida. He moved to Orlando in Fall 2007. Previously he was a professor of Computer Science at Iowa State University. His research interests include programming and specification language design and semantics, program verification, and formal methods, with an emphasis on the object-oriented and aspect-oriented paradigms. He received his Ph.D. from MIT in 1989.

Natasha Sharygina (CMU and SEI, USA; Lugano, Switzerland)
Natasha Sharygina is a senior researcher at the Carnegie Mellon Software Engineering Institute and an adjunct assistant professor in the School of Computer Science at Carnegie Mellon University, and an assistant professor at the University of Lugano. Her research interests are in program verification, formal methods in system design and analysis, systems engineering, semantics of programming languages and logics, and automated tools for reasoning about computer systems. She received her Ph.D. from The University of Texas at Austin in 2002.

SAVCBS 2007
PROGRAM COMMITTEE

	Arnd Poetzsch-Heffter (Department of CS, Univ. of Kaiserslautern) Arnd Poetzsch-Heffter chaired the program committee for SAVCBS 2007. He is a professor in the Department of Computer Science at the University of Kaiserslautern, Germany. His research interests are in component-oriented programming, program verification and generative programming. He received his Ph.D. and Habilitation Degree in Computer Science from the Technische Universität München in 1991 and 1997.

Workshop Program Committee:
Jonathan Aldrich (Carnegie Mellon University)
Michael Barnett (Microsoft Research)
Marcello M. Bonsangue (LIACS – Leiden University)
Paulo Borba (Federal University of Pernambuco)
Kathi Fisler (WPI)
Cormac Flanagan (University of California, Santa Cruz)
Marieke Huisman (INRIA Sophia Antipolis)
Joost-Pieter Katoen (RWTH Aachen)
Gary T. Leavens (Iowa State University)
Peter Müller (ETH Zürich)
David Naumann (Stevens Institute of Technology)
Matthew Parkinson (University of Cambridge)
Arnd Poetzsch-Heffter (University of Kaiserslautern), PC Chair
Ralf Reussner (Universität Karlsruhe)
Natasha Sharygina (Lugano and Carnegie Mellon)
Kurt C. Wallnau (Software Engineering Institute)
Tao Xie (North Carolina State)

SAVCBS 2007
WORKSHOP INTRODUCTION

This workshop is concerned with how formal (i.e., mathematical) techniques can be or should be used to establish a suitable foundation for the specification and verification of component-based systems. Component-based systems are a growing concern for the software engineering community. Specification and reasoning techniques are urgently needed to permit composition of systems from components. Component-based specification and verification is also vital for scaling advanced verification techniques such as extended static analysis and model checking to the size of real systems. The workshop will consider formalization of both functional and non-functional behavior, such as performance or reliability.

This workshop brings together researchers and practitioners in the areas of component-based software and formal methods to address the open problems in modular specification and verification of systems composed from components. We are interested in bridging the gap between principles and practice. The intent of bringing participants together at the workshop is to help form a community-oriented understanding of the relevant research problems and help steer formal methods research in a direction that will address the problems of component-based systems. For example, researchers in formal methods have only recently begun to study principles of object-oriented software specification and verification, but do not yet have a good handle on how inheritance can be exploited in specification and verification. Other issues are also important in the practice of component-based systems, such as concurrency, mechanization and scalability, performance (time and space), reusability, and understandability. The aim is to brainstorm about these and related topics to understand both the problems involved and how formal techniques may be useful in solving them.

The goals of the workshop are to produce:

1. An outline of collaborative research topics,
2. A list of areas for further exploration,
3. An initial taxonomy of the different dimensions along which research in the area can be categorized. For instance, static/dynamic verification, modular/whole program analysis, partial/complete specification, soundness/completeness of the analysis, are all continuums along which particular techniques can be placed, and
4. A web site that will be maintained after the workshop to act as a central clearinghouse for research in this area.

We enthusiastically thank the authors of submitted papers; their quality contributions and participation are what make a workshop like SAVCBS successful. We thank the program committee for their careful reading and reviewing of the submissions. Our PC members have expertise in a wide variety of sub-disciplines related to specification and verification of component-based systems; they include established research leaders and promising recent Ph.D.s; they come from both industry and academia, and hail from all over the world.

We received 17 submissions. All papers were reviewed by at least 3 PC members. After PC discussions via a conference tool, 8 papers were accepted for long presentation at the workshop. Similar to previous years, we accepted 6 additional submissions for short presentation, reflecting the community-building role of SAVCBS and the goal of promoting discussion and incubation of new ideas for which a full paper may be premature. One of the accepted short presentations was withdrawn by the authors. Three submissions were rejected.

This year's program also includes a solution to a specification and verification challenge problem posed to workshop attendees. The problem focused on the specification of the subject-observer pattern. This common programming pattern is to separate the component that encapsulates some state from the components that access that state. The former component is often called a *subject*, while the latter type is an *observer*. At a minimum, a subject has a method for registering an observer, a method for updating the encapsulated state, and a method for retrieving the value of the state. Observers must provide a method for being notified: the behavior of the pair is that when the update method is called, all registered observers have their notification method called. While familiar to many programmers, this problem poses real challenges for specification and verification systems and it has already been the topic of a number of papers in the field. The received and presented solution was reviewed by two members of the program committee.

Arnd Poetzsch-Heffter (Program Committee Chair)

Jonathan Aldrich (Organizing Committee)
Mike Barnett (Organizing Committee)
Dimitra Giannakopoulou (Organizing Committee)
Gary T. Leavens (Organizing Committee)
Natasha Sharygina (Organizing Committee)