

# Lecture Notes in Computer Science

2852

Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

**Springer**

*Berlin*

*Heidelberg*

*New York*

*Hong Kong*

*London*

*Milan*

*Paris*

*Tokyo*

Frank S. de Boer   Marcello M. Bonsangue  
Susanne Graf   Willem-Paul de Roever (Eds.)

# Formal Methods for Components and Objects

First International Symposium, FMCO 2002  
Leiden, The Netherlands, November 5-8, 2002  
Revised Lectures



Springer

## Series Editors

Gerhard Goos, Karlsruhe University, Germany  
Juris Hartmanis, Cornell University, NY, USA  
Jan van Leeuwen, Utrecht University, The Netherlands

## Volume Editors

Frank S. de Boer  
Centre for Mathematics and Computer Science, CWI  
Kruislaan 413, 1098 SJ Amsterdam, The Netherlands  
E-mail: frb@cw.nl

Marcello M. Bonsangue  
Leiden University, Leiden Institute of Advanced Computer Science  
P.O. Box 9512, 2300 RA Leiden, The Netherlands,  
E-mail: marcello@liacs.nl

Susanne Graf  
VERIMAG  
2 Avenue de Vignate, Centre Equitation, 38610 Grenoble-Gières, France  
E-mail: Susanne.graf@imag.fr

Willem-Paul de Rover  
Christian-Albrechts-University of Kiel  
Institute of Computer Science and Applied Mathematics  
Hermann-Rodewald-Straße 3, Kiel, Germany  
E-mail: wpr@informatik.uni-kiel.de

## Cataloging-in-Publication Data applied for

A catalog record for this book is available from the Library of Congress.

Bibliographic information published by Die Deutsche Bibliothek  
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie;  
detailed bibliographic data is available in the Internet at <<http://dnb.ddb.de>>.

CR Subject Classification (1998): D.2, D.3, F.3, D.4

ISSN 0302-9743

ISBN 3-540-20303-6 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

Springer-Verlag Berlin Heidelberg New York  
a member of BertelsmannSpringer Science+Business Media GmbH

[www.springeronline.com](http://www.springeronline.com)

© Springer-Verlag Berlin Heidelberg 2003  
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik  
Printed on acid-free paper SPIN: 10961087 06/3142 5 4 3 2 1 0

# Preface

Large and complex software systems provide the necessary infrastructure in all industries today. In order to construct such large systems in a systematic manner, the focus in the development methodologies has switched in the last two decades from functional issues to structural issues: both data and functions are encapsulated into software units that are integrated into large systems by means of various techniques supporting reusability and modifiability. This encapsulation principle is essential to both the object-oriented and the more recent component-based software engineering paradigms.

Formal methods have been applied successfully to the verification of medium-sized programs in protocol and hardware design. However, their application to large systems requires the further development of specification and verification techniques supporting the concepts of reusability and modifiability.

In order to bring together researchers and practioners in the areas of software engineering and formal methods, we organized the 1st International Symposium on Formal Methods for Components and Objects (FMCO) in Leiden, The Netherlands, November 5–8, 2002. The program consisted of invited tutorials and more technical presentations given by leading experts in the fields of Theoretical Computer Science and Software Engineering. The symposium was attended by more than 100 people.

This volume contains the contributions of the invited speakers to FMCO 2002. We believe that the presented material provides a unique combination of ideas on software engineering and formal methods which we hope will be an inspiration for those aiming at further bridging the gap between the theory and practice of software engineering.

The very idea to organize FMCO arose out of the NWO/DFG bilateral project Mobi-J. In particular we acknowledge the financial support of the NWO funding of Mobi-J. Additional financial support was provided by the Lorentz Center, the IST project Omega (2001-33522), the Dutch Institute for Programming Research and Algorithmics (IPA), the Royal Netherlands Academy of Arts and Sciences (KNAW), the Centrum voor Wiskunde en Informatica (CWI), and the Leiden Institute of Advanced Computer Science (LIACS).

July 2003

F.S. de Boer  
M.M. Bonsangue  
S. Graf  
W.-P. de Roever  
(Editors)

## The Mobi-J Project

Mobi-J is a project founded by a bilateral research program of the Dutch Organization for Scientific Research (NWO) and the Central Public Funding Organization for Academic Research in Germany (DFG).

The partners of the Mobi-J projects are:

- Centrum voor Wiskunde en Informatica (F.S. de Boer)
- Leiden Institute of Advanced Computer Science (M.M. Bonsangue)
- Christian-Albrechts-Universität, Kiel (W.-P. de Roever)

This project aims at the development of a programming environment which supports component-based design and verification of Java programs annotated with assertions. The overall approach is based on an extension of the Java language called Mobi-J with the notion of a component which provides for the encapsulation of its internal processing of data and composition in a network by means of mobile asynchronous channels.

The activities of Mobi-J include the organization of international symposia funded by the NWO and Ph.D. research funded by DFG. By means of regular meetings the partners discuss intensively Ph.D. research involving Mobi-J-related topics. Mobi-J also maintains contacts with other German universities, including the universities of Oldenburg and Munich, and a close collaboration with the European IST project OMEGA.

## The Omega Project

The overall aim of the European IST project Omega (2001-33522) is the definition of a development methodology in UML for embedded and real-time systems based on formal verification techniques. The approach is based on a formal semantics of a suitable subset of UML, adapted and extended where needed with a special emphasis on time-related aspects.

The Omega project involves the following partners:

VERIMAG (France, Coordinator)  
 Centrum voor Wiskunde en Informatica (The Netherlands)  
 Christian-Albrechts-Universität (Germany)  
 University of Nijmegen (The Netherlands)  
 Wiezmann Institute (Israel)  
 OFFIS (Germany),  
 EADS Launch Vehicles (France)  
 France Telecom R&D (France)  
 Israel Aircraft Industries (Israel)  
 National Aerospace Laboratory (The Netherlands)

# Table of Contents

A Tool-Supported Proof System for Multithreaded Java .....	1
<i>E. Abraham, F.S. de Boer, W.-P. de Roever, and M. Steffen</i>	
Abstract Behavior Types: A Foundation Model for Components and Their Composition .....	33
<i>F. Arbab</i>	
Understanding UML: A Formal Semantics of Concurrency and Communication in Real-Time UML .....	71
<i>W. Damm, B. Josko, A. Pnueli, and A. Votintseva</i>	
Live and Let Die: LSC-Based Verification of UML-Models .....	99
<i>W. Damm and B. Westphal</i>	
Reactive Animation .....	136
<i>D. Harel, S. Efroni, and I.R. Cohen</i>	
Model-Checking Middleware-Based Event-Driven Real-Time Embedded Software .....	154
<i>X. Deng, M.B. Dwyer, J. Hatcliff, G. Jung, Robby, and G. Singh</i>	
Equivalent Semantic Models for a Distributed Dataspace Architecture ....	182
<i>J. Hooman and Jaco van de Pol</i>	
Java Program Verification Challenges .....	202
<i>B. Jacobs, J. Kiniry, and M. Warnier</i>	
ToolBus: The Next Generation .....	220
<i>H. de Jong and P. Klint</i>	
High-Level Specifications: Lessons from Industry .....	242
<i>B. Batson and L. Lamport</i>	
How the Design of JML Accommodates Both Runtime Assertion Checking and Formal Verification .....	262
<i>G.T. Leavens, Y. Cheon, C. Clifton, C. Ruby, and D.R. Cok</i>	
Finding Implicit Contracts in .NET Components .....	285
<i>K. Arnout and B. Meyer</i>	
From Co-algebraic Specifications to Implementation: The Mihda Toolkit ..	319
<i>G. Ferrari, U. Montanari, R. Raggi, and E. Tuosto</i>	
A Calculus for Modeling Software Components .....	339
<i>O. Nierstrasz and F. Achemann</i>	

Specification and Inheritance in CSP-OZ ..... 361  
*E.-R. Olderog and H. Wehrheim*

Model-Based Testing of Object-Oriented Systems ..... 380  
*B. Rumpe*

Concurrent Object-Oriented Programs: From Specification to Code..... 403  
*E. Sekerinski*

Design with Asynchronously Communicating Components..... 424  
*J. Plosila, K. Sere, and M. Waldén*

Composition for Component-Based Modeling ..... 443  
*G. Gössler and J. Sifakis*

Games for UML Software Design..... 467  
*P. Stevens and J. Tenzer*

Making Components Move: A Separation of Concerns Approach ..... 487  
*D. Pattinson and M. Wirsing*

**Author Index** ..... 509