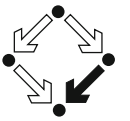
Texts and Monographs in Symbolic Computation

A Series of the Research Institute
for Symbolic Computation,
Johannes Kepler University, Linz, Austria



More information about this series at
http://www.springer.com/series/3073

Bernhard Thalheim • Klaus-Dieter Schewe •
Andreas Prinz • Bruno Buchberger
Editors

Correct Software in Web Applications and Web Services

Springer

Editors
Bernhard Thalheim
Institut für Informatik
Christian-Albrechts-Universität
Kiel
Germany

Klaus-Dieter Schewe
Software Competence Center
Hagenberg
Austria

Andreas Prinz
ICT Department
University of Agder
Kristiansand
Norway

Bruno Buchberger
RISC
Johannes Kepler University
Hagenberg
Austria

ISSN 0943-853X ISSN 2197-8409 (electronic)
Texts and Monographs in Symbolic Computation
ISBN 978-3-319-17111-1 ISBN 978-3-319-17112-8 (eBook)
DOI 10.1007/978-3-319-17112-8

Library of Congress Control Number: 2015940569

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

This volume constitutes selected and extended papers of a European Science Foundation (ESF) strategic workshop.

The workshop "Correct Software in Web Applications" was held at the Research Institute for Symbolic Computation (RISC), part of the Johannes Kepler University Linz in Hagenberg, Austria, over 3 days. Originally, 30 participants from 12 countries have been invited in September 2011. Twenty-six researchers from Austria, Germany, Great Britain, Hungary, Italy, France, Norway and Romania participated in the workshop. Fourteen researchers were invited for submission of an extended research paper. Finally, we selected nine from these papers for this volume.

The workshop programme consisted of presentations and intensive discussion rounds. The presentations covered abstract state machines (ASMs) and Event-B as formal methods and experiences in applying them to selected fields of Web applications, Theorema and Karlsruhe Interactive Verifier (KIV) as verification methods and experiences made with these tools for Web applications and various detailed descriptions of facets of the large field of Web applications such as Web information systems with emphasis on storyboarding and media types, recommender systems, common protocols such as Hypertext Transfer Protocol (HTTP), browser technology, scripting technology, security aspects of Web services and Web services orchestration. The three discussion sessions addressed the characterisation of what constitutes a Web application, what does correctness mean in this context and what is a common umbrella for a research agenda in this field.

Web applications have become one of the largest area for the application of software engineering methods. At the beginning, Web applications were only simple information services, which soon developed into large database-backed Web information systems, i.e. data-intensive systems that are accessed and maintained via the World Wide Web. More recently, the area of Web services has emerged, referring to software components on some Web servers that can be accessed by and integrated into other software systems. There is a tendency to extend Web information systems and Web services towards large-scale interoperable systems.

This marks a shift towards computation in the public domain using the Internet as the medium for interaction of software components.

Despite the immense importance of Web applications for software engineering, there is a lack of well-founded development methods. Surprisingly, many Web applications are created in an ad hoc manner without the use of sophisticated formal methods. Quality assurance is to a large extent ignored. This constitutes a barrier to productive software development; in other words, the envisioned and most likely technically possible shift to computation in the public domain will only become reality if the resulting systems are trustworthy with respect to consistency, reliability, performance and security. Thus, there is a need for development methods that lead to provably correct Web application systems.

The world and correspondingly information technology (IT) are continuously changing. Systems become accessible through networks. The corresponding infrastructure exists, emerges and provides all what is necessary for cooperations. Therefore, applications can run on systems that are rented by companies. Cloud systems are based on new paradigms such as software-as-a-service, data-as-a-service and infrastructure-as-a-service. Services systems are currently far more advanced than the collaboration systems in the past. At the same time, the theory and conception of a service are not yet properly developed. High-quality services are needed. But despite the fact that services are already an essential part of nowadays IT infrastructures, there remain significant lacunas in our understanding of what services are and of how they work—services are correct and they deliver completely and only what has been asked for—and of proper development of correct and reliable service infrastructures.

Abstract state machines (ASMs) provide a general method to combine specifications on any desired level of abstraction, ground modelling (requirement capture) techniques and stepwise refinement to executable code, providing the basis for experimental validation and mathematical verification. ASMs have been successfully applied to diverse areas such as specification and verification of the implementation of programming languages (e.g. Prolog2WAM, Occam2Transputer, Java2JVM, C#2CLR) and of chip design, train control systems, the Mondex electronic purse and many more. These success stories involve verification by mathematical proofs as well as proofs by theorem provers (e.g. KIV, Prototype Verification System (PVS), Isabelle) or model checking with justifiable effort.

The key problem addressed by the proposed ESF explorative workshop is that there is almost no connection between the research on these industrially successful formal methods in software engineering and the important area of Web applications. Researchers in symbolic computation, abstract state machines, automated reasoning and verification need input from researchers in Web information systems, Web services, interoperability and service-oriented architectures to tailor their research to the needs of the applications, and researchers in Web applications engineering need support to address the challenging correctness problems.

This volume aims to:

1. Obtain a common understanding of the challenging research questions in Web applications comprising Web information systems, Web services and Web interoperability
2. Obtain a common understanding of verification needs in Web applications
3. Achieve a common understanding of the available rigorous approaches to system development and the cases in which they succeeded
4. Identify how rigorous software engineering methods can be exploited to develop correct Web applications
5. Develop a European scale research agenda comprising theory, methods and tools that would lead to correct Web applications with the potential to realise systems for computation in the public domain
6. Develop a formal model of services and facilities for analysis, control and test of such services

The main results are as follows:

- An identification of correctness problems in Web applications and sketches and how these can be solved by formalised software engineering methods, in particular Theorema and ASMs
- An identification of open problems regarding correctness of Web applications and corresponding research questions that have to be addressed in the context of Theorema and ASMs to solve these problems
- A common understanding regarding the need for assuring correctness and the potential of formalised methods with this regard
- A vision for a research agenda, preferably grouped into project topics, to address the open problems regarding correctness of Web applications
- A proposal for a research agenda and a commented list of open problems

Idir Ait-Sadoune and Yamine Ait-Ameur propose an extension of the Business Process Execution Language (BPEL) on the basis of Event-B semantics for formal modelling of Web services compositions that covers the scope, the fault and the compensation handlers. A resulting methodology properly supports the design of transactional BPEL processes. The proposed approach is illustrated by a case study.

Maria Bergholtz, Birger Andersson and Paul Johannesson develop a model of services that allows to properly specify and to analyse the concept of a service based on an understanding of services as a means for cocreation of value, as a means for abstraction and as a means for distributing rights.

Marian Borek, Kuzman Katkalov, Nina Moebius, Wolfgang Reif, Gerhard Schellhorn and Kurt Stenzel develop a development method for secure service applications that integrates a model-driven approach with formal specification techniques using abstract state machines, refinement to code and verification with the interactive theorem prover KIV.

Károly Bósa, Roxana Holom and Mircea Boris Vleju propose a uniform client-cloud interaction approach by which cloud service owners are able to fully control the usages of their services in the case of each user subscription. The applied

method is able to incorporate the major advantages of the ASMs and of ambient calculus.

Ajantha Dahanayake and Bernhard Thalheim develop a conceptual model for service specification based on a general model framework W*H that extends the rhetoric frame by Hermagoras of Temnos. The framework separates service concerns such as service as a product, service as an offer, service request, service delivery, service application, service record, service log or archive and service exception.

Harald Lampesberger and Mariam Rady show how monitoring is complementing testing and formal methods for Web and cloud systems. The approach extends service-level agreement based on negotiations between clients and providers and thus supports analysis of correctness of the interaction.

Raffaela Mirandola, Pasqualina Potena, Elvinia Riccobene and Patrizia Scandurra present two approaches to predict and analyse reliability of a Web service based on BPEL from one side and on SCA-ASM from the other side. It is shown that the second approach is more effective in comparison with the first one.

Klaus-Dieter Schewe and Qing Wang propose a theory of services ($BDCM^2$) that captures behaviour, description, contracting, monitoring and mediation based on abstract state services, on service mediators and on service ontology models.

Bernhard Thalheim and Klaus-Dieter Schewe survey the codesign approach for integrated development of structuring, functionality, distribution and interactivity for Web information systems. The specification framework has been applied for design and realisation of large information-intensive e-business, edutainment (e-learning), infotainment and community websites.

Reviewers. We thank our reviewers for their efforts, for their detailed reviews and for the support for their second round of reviewing revised papers:

Yamine Ait Ameur
Birger Anderson
Maria Bergholtz
Marian Borek
Karoly Bosa
Ajantha Dahanayake
Antje Düsterhöft
Roxana Holom
Paul Johannesson
Kuzman Katkalov
Meike Klettke
Frank Kramer
Harald Lampesberger
Hui Ma
Raffaela Mirandola
Nina Möbius
Pascalina Potena
Andreas Prinz
Miriam Rady
Wolfgang Reif
Elvinia Riccobene
Patrizia Scandurra
Gerhard Schellhorn
Klaus-Dieter Schewe
Ove Sörensen
Kurt Stenzel
Bernhard Thalheim
Marina Tropmann
Boris Vleju
Qing Wang

Finally, we thank the Springer team for their help, their support and their patience, especially to Silvia Schilgerius.

Kiel, Germany — Bernhard Thalheim
Hagenberg, Austria — Klaus-Dieter Schewe
Kristiansand, Norway — Andreas Prinz
Hagenberg, Austria — Bruno Buchberger

Contents

Contributors

Yamine Ait-Ameur IRIT - ENSEEIHT, Toulouse, France

Idir Ait-Sadoune LRI - CentraleSupelec, Gif-Sur-Yvette, France

Birger Andersson Department of Computer and Systems Sciences, Stockholm University, Kista, Sweden

Maria Bergholtz Department of Computer and Systems Sciences, Stockholm University, Kista, Sweden

Marian Borek Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Károly Bósa Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Roxana Chelemen Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Ajantha Dahanayake Department of Computer Information Science, Prince Sultan University, Riyadh, Kingdom of Saudi Arabia

Paul Johannesson Department of Computer and Systems Sciences, Stockholm University, Kista, Sweden

Kuzman Katkalov Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Harald Lampesberger Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Raffaela Mirandola Politecnico di Milano, Milano, Italy

Nina Moebius Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Pasqualina Potena Università degli Studi di Bergamo, Dalmine (BG), Italy

Mariam Rady Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Wolfgang Reif Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Elvinia Riccobene Università degli Studi di Milano, Crema, Italy

Patrizia Scandurra Università degli Studi di Bergamo, Dalmine (BG), Italy

Gerhard Schellhorn Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Klaus-Dieter Schewe Software Competence Center Hagenberg, Hagenberg, Austria
and
Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Kurt Stenzel Institute for Software and Systems Engineering, Augsburg University, Augsburg, Germany

Bernhard Thalheim Department of Computer Science, Christian Albrechts University Kiel, Kiel, Germany

Mircea Boris Vleju Christian Doppler Laboratory for Client-Centric Cloud Computing, Johannes Kepler University Linz, Hagenberg, Austria

Qing Wang Research School of Computer Science, The Australian National University, Canberra, ACT, Australia