

---

# Perspectives on the Future of Software Engineering



Jürgen Münch • Klaus Schmid  
Editors

# Perspectives on the Future of Software Engineering

Essays in Honor of Dieter Rombach



Springer

*Editors*

Jürgen Münch  
University of Helsinki  
Department of Computer Science  
Helsinki  
Finland

Klaus Schmid  
Universität Hildesheim  
Institut für Informatik  
Hildesheim  
Germany

ISBN 978-3-642-37394-7      ISBN 978-3-642-37395-4 (eBook)  
DOI 10.1007/978-3-642-37395-4  
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013939055

ACM Codes: D.2, K.6

© Springer-Verlag Berlin Heidelberg 2013

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))



Prof. Dr. Dr. h. c. Dieter Rombach



---

## Preface

Our modern society and technology are built on software. Critical embedded systems such as cars or factories, information systems such as ERP solutions or Internet search engines, or infrastructure such as utilities or telecommunications – none would work for even a second were it not for software. Most innovations today are shaped by software and nearly all businesses and industries are transformed by software. We can be quite sure that this will not change and software will continue to be at the core of major future changes. This relevance and dependence on quality software is what makes *software engineering* a key discipline for modern society. Software engineering is the discipline that aims at providing, evaluating, and improving methods, techniques, processes, and tools for the development of defect-free software that fulfills the needs of customers and users within time and budget constraints. Along with the growing importance of software, software engineering has also become a core field of modern research.

Since its inception in the 1960s, software engineering as a discipline has constantly grown and matured in many areas and in many ways. Today, it is a rich discipline with well-established research methods, consisting of many different subdisciplines. Of key importance to the development of a research discipline is always the underlying scientific approach. Here, the discipline has seen a major shift over the last three decades, as its formal foundations were successively augmented by a focus on empirical work aimed at evaluating whether research approaches do contribute value in real-world situations. This came to be known as empirical software engineering and is an important component of any modern software engineering research. One of the leading protagonists of empirical software engineering worldwide and certainly the leader in Germany in this subject is Dieter Rombach.

Prof. Dr. Dr. h.c. Dieter Rombach dedicated his entire career to furthering the cause of empirical software engineering as a discipline. In particular, his main research interests have always been in developing software with predictable quality. He has done intensive work on quantitative methods, languages, and tools to support software process and project management. This focus already became visible while he was working on his dissertation at the University of Kaiserslautern, Germany, where he conducted one of the largest controlled experiments ever – the development and maintenance of eight operating system kernels – to provide evidence regarding the benefits of a new structuring concept for maintainability. Ph.D. degree

in hand, he went to spend several years at the University of Maryland, where he worked with Victor Basili, who can be regarded as one of the founding fathers of empirical software engineering research. During this time, Dieter Rombach also worked as a project leader at the NASA Goddard Space Flight Center. In 1990, he received the Presidential Young Investigator Award of the National Science Foundation (NSF), USA. In 1992, he returned to the University of Kaiserslautern. His strong dedication to industrial cooperation directly led to the foundation of the Software Technologie Transfer Initiative (STTI), which later resulted in the creation of the Fraunhofer Institute for Experimental Software Engineering (IESE). This institute currently has about 200 employees and has been an important contributor to the international software engineering world for more than 15 years, exerting a strong influence in both research and industry. In particular, it has helped to significantly promote the concept of empirical software engineering. Beyond his personal impact and the impact of the institute he leads, he has achieved significant impact indirectly through the many students he has advised and taught over the years, including about 60 Ph.D. students.

For his many and important contributions to the field, Dieter Rombach has received numerous awards and recognitions, like the Service Medal of the State of Rhineland-Palatinate and the Federal Cross of Merit on Ribbon of the Federal Republic of Germany. He has also received an honorary doctorate degree from the University of Oulu, Finland, and was elected a Fellow by both the ACM and the IEEE Computer Society.

His impact on the software engineering landscape is amplified by his role as an expert, reviewer, and consultant to industry and as an advisor to different state, federal, and international bodies. Instead of going into more detail on his many achievements, we refer the reader to his bio.<sup>1</sup> Some things, however, cannot be found there, like his strong dedication to Kaiserslautern, his commitment to his favorite soccer team 1. FC Kaiserslautern, and many other things. They show that, while he spends numerous hours on software engineering, his interests are much broader.

This book is dedicated to Dieter Rombach and his contributions to software engineering in general and to empirical software engineering in particular. In fact, it was written to accompany a symposium in honor of his 60th birthday. But beyond this, its aim is to take stock of the current situation in software engineering and point out some visions for the future. This aim guided the concept of this book throughout. We introduce the book with a paper written by Dieter Rombach that provides a good overview of his vision for the empirical software engineering discipline. The remainder of the book is structured into three main parts: The first part focuses on what are generally considered the classical foundations of software engineering research, such as notations, architecture, and processes. The second part addresses the core part of Dieter Rombach's contribution – empirical software engineering – while the third part discusses the broader vision of the software engineering discipline, described along various dimensions. Contributions to this

---

<sup>1</sup>[http://www.iese.fraunhofer.de/en/aboutus/directors/cv\\_rombach\\_english.html](http://www.iese.fraunhofer.de/en/aboutus/directors/cv_rombach_english.html)



volume were collected on a by-invitation basis only. Invitations were sent to selected, internationally renowned researchers who have a relationship with Dieter Rombach's work and history. Due to the enormous network of collaborations that he has created over the years, the latter was hardly a restriction. Most of the authors invited promised a contribution right away, which now forms part of this collection. We are very happy about the numerous internationally acclaimed authors who did not hesitate to contribute to this collection. Without their contributions, this book would just not have been possible!

We augmented the collection with contributions by current members of Fraunhofer IESE to ensure that the research focus of Dieter Rombach, which is embedded in Fraunhofer IESE today, is adequately represented throughout this collection. As a result, we believe that this collection now provides an excellent overview of the current state of software engineering and its future directions and emphasizes the specific influences by Dieter Rombach and the research he cares about most.

A collection like this would never be possible without the help of many people. First of all, we would like to thank the numerous authors for their contributions. We know that it is not easy to make room in a busy schedule to be able to write profound contributions like the ones we received for this book, particularly within a tight schedule. The collaboration was simply exceptional! We would also like to thank Fraunhofer IESE as the sponsor of this book and several of its staff who greatly helped in preparing the book: Mrs. Nicole Spanier-Baro, who worked on the administrative issues and the accompanying symposium; Ms. Sonnhild Namingha, who did a great job of proofreading and editing; and Stephan Thiel, who worked relentlessly to get all the final formatting work done. We would also like to thank Christian Kröher from the University of Hildesheim for supporting us with LaTeX editing and Ralf Gerstner from Springer, who worked on the contract issues and supported us at every turn. Finally, we are grateful to Martin Verlage, who worked with us on the concept of the book and contributed a lot of ideas to our discussions.

Helsinki, Finland  
Hildesheim, Germany

Jürgen Münch  
Klaus Schmid



---

# Contents

<b>Empirical Software Engineering Models: Can They Become the Equivalent of Physical Laws in Traditional Engineering? .....</b>	<b>1</b>
Dieter Rombach	
<b>Part I Software Development: Notation, Architecture, and Process</b>	
<b>Domain Modeling and Domain Engineering: Key Tasks in Requirements Engineering .....</b>	<b>15</b>
Manfred Broy	
<b>Towards Agile Verification .....</b>	<b>31</b>
Carlo Ghezzi, Amir Molzam Sharifloo, and Claudio Menghi	
<b>On Model-Based Software Development .....</b>	<b>49</b>
Constance L. Heitmeyer, Sandeep Shukla, Myla M. Archer, and Elizabeth I. Leonard	
<b>From Software Systems to Complex Software Ecosystems: Model- and Constraint-Based Engineering of Ecosystems .....</b>	<b>61</b>
Andreas Rausch, Christian Bartelt, Sebastian Herold, Holger Klus, and Dirk Niebuhr	
<b>A Safety Roadmap to Cyber-Physical Systems .....</b>	<b>81</b>
Mario Trapp, Daniel Schneider, and Peter Liggesmeyer	
<b>Modeling Complex Information Systems .....</b>	<b>95</b>
Joerg Doerr	
<b>Continuous Process Improvement .....</b>	<b>111</b>
Jens Heidrich	
<b>Part II Empirical Research and Studies</b>	
<b>Paths to Software Engineering Evidence .....</b>	<b>133</b>
Ross Jeffery	

<b>An Evidence Profile for Software Engineering Research and Practice</b> ....	145
Claes Wohlin	
<b>Challenges of Evaluating the Quality of Software Engineering Experiments</b> .....	159
Oscar Dieste and Natalia Juristo	
<b>Technical Debt: Showing the Way for Better Transfer of Empirical Results</b> .....	179
Forrest Shull, Davide Falessi, Carolyn Seaman, Madeline Diep, and Lucas Layman	
<b>An Empirical Investigation of the Component-Based Performance Prediction Method Palladio</b> .....	191
Ralf Reussner, Steffen Becker, Anne Koziolok, and Heiko Koziolok	
<b>Can We Trust Software Repositories?</b> .....	209
Andreas Zeller	
<b>Empirical Practice in Software Engineering</b> .....	217
Andreas Jedlitschka, Liliana Guzmán, Jessica Jung, Constanza Lampasona, and Silke Steinbach	
<b>Part III Visions on the Future of Software Engineering as a Discipline</b>	
<b>What Is Software? The Role of Empirical Methods in Answering the Question</b> .....	237
Leon J. Osterweil	
<b>A Personal Perspective on the Evolution of Empirical Software Engineering</b> .....	255
Victor R. Basili	
<b>Moving Toward Evidence-Based Software Production</b> .....	275
David M. Weiss, James Kirby Jr., and Robyn R. Lutz	
<b>Skating to Where the Puck Is Going: Future Systems and Software Engineering Opportunities and Challenges</b> .....	299
Barry Boehm	
<b>Formalism and Intuition in Software Engineering</b> .....	335
Michael Jackson	
<b>Education of Software Engineers</b> .....	349
Marvin V. Zelkowitz	
<b>Integrated Software Process and Product Lines</b> .....	359
Dieter Rombach	

---

## Contributors

**Myla M. Archer** Naval Research Laboratory, Software Engineering, Washington, DC, USA, [archer@itd.nrl.navy.mil](mailto:archer@itd.nrl.navy.mil)

**Christian Bartelt** Chair of Software Systems Engineering, Department of Informatics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, [christian.bartelt@tu-clausthal.de](mailto:christian.bartelt@tu-clausthal.de)

**Victor R. Basili** Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD, USA

Fraunhofer Center for Experimental Software Engineering, College Park, USA, [basili@cs.umd.edu](mailto:basili@cs.umd.edu)

**Steffen Becker** Fachgruppe Softwaretechnik, Heinz Nixdorf Institut, Universität Paderborn, Paderborn, Germany, [steffen.becker@uni-paderborn.de](mailto:steffen.becker@uni-paderborn.de)

**Barry Boehm** University of Southern California, Los Angeles, CA, USA, [boehm@usc.edu](mailto:boehm@usc.edu)

**Manfred Broy** Institut für Informatik, Technische Universität München, München, Germany, [broy@in.tum.de](mailto:broy@in.tum.de)

**Madeline Diep** Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA, [mdiep@fc-md.umd.edu](mailto:mdiep@fc-md.umd.edu)

**Oscar Dieste** Universidad Politécnica de Madrid, Madrid, Spain, [odieste@fi.upm.es](mailto:odieste@fi.upm.es)

**Jörg Dörr** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Joerg.Doerr@iese.fraunhofer.de](mailto:Joerg.Doerr@iese.fraunhofer.de)

**Davide Falessi** Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA, [dfalessi@fc-md.umd.edu](mailto:dfalessi@fc-md.umd.edu)

**Carlo Ghezzi** Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy, [carlo.ghezzi@polimi.it](mailto:carlo.ghezzi@polimi.it)

**Liliana Guzmán** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [liliana.guzmn@iese.fraunhofer.de](mailto:liliana.guzmn@iese.fraunhofer.de)

**Jens Heidrich** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Jens.Heidrich@iese.fraunhofer.de](mailto:Jens.Heidrich@iese.fraunhofer.de)

**Constance L. Heitmeyer** Naval Research Laboratory, Software Engineering, Washington, DC, USA, [heimtyme@itd.nrl.navy.mil](mailto:heimtyme@itd.nrl.navy.mil)

**Sebastian Herold** Chair of Software Systems Engineering, Department of Informatics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, [sebastian.herold@tu-clausthal.de](mailto:sebastian.herold@tu-clausthal.de)

**Michael Jackson** Department of Computing, The Open University, Milton Keynes, UK, [jacksonma@acm.org](mailto:jacksonma@acm.org)

**Andreas Jedlitschka** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [andreas.jedlitschka@iese.fraunhofer.de](mailto:andreas.jedlitschka@iese.fraunhofer.de)

**Ross Jeffery** NICTA, Eveleigh, NSW, Australia

School of Computer Science and Engineering, University of New South Wales, Kensington, Australia, [ross.jeffery@nicta.com.au](mailto:ross.jeffery@nicta.com.au)

**Jessica Jung** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [jessica.jung@iese.fraunhofer.de](mailto:jessica.jung@iese.fraunhofer.de)

**Natalia Juristo** Universidad Politécnica de Madrid, Madrid, Spain, [natalia@fi.upm.es](mailto:natalia@fi.upm.es)

**James Kirby Jr.** Naval Research Laboratory, Washington, DC, USA, [james.kirby@nrl.navy.mil](mailto:james.kirby@nrl.navy.mil)

**Holger Klus** Chair of Software Systems Engineering, Department of Informatics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, [holg.klus@tu-clausthal.de](mailto:holg.klus@tu-clausthal.de)

**Anne Koziolk** Institut für Programmstrukturen und Datenorganisation (IPD), Karlsruhe Institut für Technologie (KIT), Karlsruhe, Germany, [anne.koziolk@kit.edu](mailto:anne.koziolk@kit.edu)

**Heiko Koziolk** ABB Corporate Research, Ladenburg, Germany, [heiko.koziolk@de.abb.com](mailto:heiko.koziolk@de.abb.com)

**Constanza Lampasona** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [constanza.lampasona@iese.fraunhofer.de](mailto:constanza.lampasona@iese.fraunhofer.de)

**Lucas Layman** Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA, [llayman@fc-md.umd.edu](mailto:llayman@fc-md.umd.edu)

**Elizabeth I. Leonard** Naval Research Laboratory, Software Engineering, Washington, DC, USA, [leonard@itd.nrl.navy.mil](mailto:leonard@itd.nrl.navy.mil)

**Peter Liggesmeyer** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Peter.Liggesmeyer@iese.fraunhofer.de](mailto:Peter.Liggesmeyer@iese.fraunhofer.de)

**Robyn R. Lutz** Department of Computer Science, Ames, IA, USA,  
[rlutz@cs.iastate.edu](mailto:rlutz@cs.iastate.edu)

**Claudio Menghi** Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy, [claudio1.menghi@mail.polimi.it](mailto:claudio1.menghi@mail.polimi.it)

**Jürgen Münch** Department of Computer Science, University of Helsinki, Helsinki, Finland, [juergen.muench@cs.helsinki.fi](mailto:juergen.muench@cs.helsinki.fi)

**Dirk Niebuhr** Chair of Software Systems Engineering, Department of Informatics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, [dirk.niebuhr@tu-clausthal.de](mailto:dirk.niebuhr@tu-clausthal.de)

**Leon J. Osterweil** Laboratory for Advanced Software Engineering Research, School of Computer Science, University of Massachusetts, Amherst, MA, USA, [ljo@cs.umass.edu](mailto:ljo@cs.umass.edu)

**Andreas Rausch** Chair of Software Systems Engineering, Department of Informatics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, [andreas.rausch@tu-clausthal.de](mailto:andreas.rausch@tu-clausthal.de)

**Ralf Reussner** Institut für Programmstrukturen und Datenorganisation (IPD), Karlsruhe Institut für Technologie (KIT), Karlsruhe, Germany, [ralf.reussner@kit.edu](mailto:ralf.reussner@kit.edu)

**H. Dieter Rombach** Technische Universität Kaiserslautern, Kaiserslautern, Germany

Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Dieter.Rombach@iese.fraunhofer.de](mailto:Dieter.Rombach@iese.fraunhofer.de)

**Klaus Schmid** Software Systems Engineering, Institute of Computer Science, University of Hildesheim, Hildesheim, Germany, [schmid@sse.uni-hildesheim.de](mailto:schmid@sse.uni-hildesheim.de)

**Daniel Schneider** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Daniel.Schneider@iese.fraunhofer.de](mailto:Daniel.Schneider@iese.fraunhofer.de)

**Carolyn Seaman** Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA

University of Maryland, Baltimore County, USA

Department of Information Systems, Information Technology & Engineering (ITE) Building, Baltimore, MD, USA, [cseaman@fc-md.umd.edu](mailto:cseaman@fc-md.umd.edu); [cseaman@umbc.edu](mailto:cseaman@umbc.edu)

**Amir Molzam Sharifloo** Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy, [amir.molzam@mail.polimi.it](mailto:amir.molzam@mail.polimi.it)

**Sandeep Shukla** Virginia Tech, Arlington Research Center, Arlington, VA, USA, [shukla@vt.edu](mailto:shukla@vt.edu)

**Forrest Shull** Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA, [fshull@fc-md.umd.edu](mailto:fshull@fc-md.umd.edu)

---

**Silke Steinbach** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [silke.steinbach@iese.fraunhofer.de](mailto:silke.steinbach@iese.fraunhofer.de)

**Mario Trapp** Fraunhofer Institut for Experimental Software Engineering (IESE), Kaiserslautern, Germany, [Mario.Trapp@iese.fraunhofer.de](mailto:Mario.Trapp@iese.fraunhofer.de)

**David M. Weiss** Department of Computer Science, Lanh & Oanh Professor of Software Engineering, Ames, IA, USA, [weiss@iastate.edu](mailto:weiss@iastate.edu)

**Claes Wohlin** School of Computing, Blekinge Institute of Technology, Karlskrona, Sweden, [Claes.Wohlin@bth.se](mailto:Claes.Wohlin@bth.se)

**Marvin V. Zelkowitz** Computer Science, University of Maryland, College Park, MD, USA

Fraunhofer Center for Experimental Software Engineering, College Park, MD, USA, [mvz@cs.umd.edu](mailto:mvz@cs.umd.edu)

**Andreas Zeller** Software Engineering Chair, Saarland University – Computer Science, Saarbrücken, Germany, [zeller@cs.uni-saarland.de](mailto:zeller@cs.uni-saarland.de)