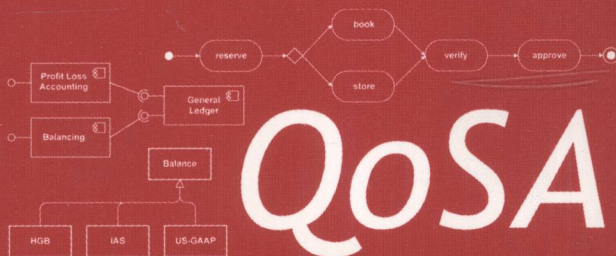


LNCS 4214

Christine Hofmeister
Ivica Crnkovic
Ralf Reussner (Eds.)

Quality of Software Architectures

Second International Conference
on Quality of Software Architectures, QoSA 2006
Västerås, Sweden, June 2006, Revised Papers



QoSA



Springer

TP311.5-53

Q1

2006

Christine Hofmeister Ivica Crnkovic
Ralf Reussner (Eds.)

Quality of Software Architectures

Second International Conference
on Quality of Software Architectures, QoSA 2006
Västerås, Sweden, June 27-29, 2006
Revised Papers



Springer



E2007001420

Volume Editors

Christine Hofmeister
Computer Science and Engineering Department
Lehigh University
Bethlehem, Pennsylvania 18015, USA
E-mail: crh@cse.lehigh.edu

Ivica Crnkovic
Software Engineering Lab
Mälardalen University
SE-721 23 Västerås, Sweden
E-mail: ivica.crnkovic@mdh.se

Ralf Reussner
Institute for Program Structures and Data Organization
University of Karlsruhe (TH)
Karlsruhe, Germany
E-mail: reussner@ipd.uka.de

Library of Congress Control Number: 2006937334

CR Subject Classification (1998): D.2.4, F.3, D.4, C.4, K.4.4, C.2

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-540-48819-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-48819-4 Springer 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. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springer.com

© Springer-Verlag Berlin Heidelberg 2006
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11921998 06/3142 5 4 3 2 1 0

Preface

Although the quality of a system's software architecture is one of the critical factors in its overall quality, the architecture is simply a means to an end, the end being the implemented system. Thus the ultimate measure of the quality of the software architecture lies in the implemented system, in how well it satisfies the system and project requirements and constraints and whether it can be maintained and evolved successfully. In order to treat design as a science rather than an art, we need to be able to address the quality of the software architecture directly, not simply as it is reflected in the implemented system.

Therefore, QoSA is concerned with software architecture quality directly by addressing the problems of:

- Designing software architectures of good quality
- Defining, measuring, evaluating architecture quality
- Managing architecture quality, tying it upstream to requirements and downstream to implementation, and preserving architecture quality throughout the lifetime of the system

Cross-cutting these problems is the question of the nature of software architecture. Software architecture organizes a system, partitioning it into elements and defining relationships among the elements. For this we often use multiple views, each with a different organizing principle.

But software architecture must also support properties that are emergent, that cannot be ascribed to particular elements. For this we often use the language of quality attributes. Quality attributes cover both internal properties, exhibited only in the development process (e.g., maintainability, portability, testability, etc.), and external properties, exhibited in the executing system (e.g., performance, resource consumption, availability, etc.). Quality attributes cover properties that are emergent, that have a pervasive impact, that are difficult to reverse, and that interact, thereby precluding or constraining other properties.

Thus in addition to examining software architecture quality, QoSA also aims to investigate quality attributes in the context of the problems of the design, evaluation, and management of software architecture. The papers selected for QoSA 2006 describe research and experience on these topics. Architecture evaluation is the most prevalent theme of the papers. The approaches vary from formal models to support evaluation to experience with process-centered approaches. The focus of the evaluation varies from evaluation of a particular quality attribute, such as performance or safety, to approaches where the evaluation covers a number of quality attributes, determined by the evaluator. Other themes for QoSA 2006 were processes for achieving, supporting and ensuring architecture quality. These papers go beyond the problem of evaluation to address software architecture quality at the process level. A final significant theme is the problem of managing and applying architectural knowledge.

Of the 30 papers submitted, 12 were selected as papers for this post-conference proceedings volume. A number of shorter papers describing emerging results or case studies were also presented at QoSA; these papers are published as technical report No 2006/10, of the University of Karlsruhe, on Perspectives in Software Architecture Quality.

As a part of the QoSA conference, a special “Industrial Day” event was organized. This included two inspiring keynote presentations by Jan Bosch and Clemens Szyperski, three tutorials, and a panel in which representatives from several international software-intensive companies participated. The abstracts of the keynotes and tutorials are available in this proceedings volume.

We thank the members of the Program Committee and additional reviewers for their thorough, thoughtful, and timely reviews of the submitted papers. We thank Steffen Becker, Sven Overhage, and Judith Stafford for their work in supporting QoSA and Klaus Krogmann for preparing this LNCS proceedings volume. Finally, we thank the generous sponsors of QoSA 2006: University of Karlsruhe (TH), Mälardalen University, and Västerås City. This conference would not be possible without the support of all the above people and sponsors.

September 2006

Ivica Crnkovic
Christine Hofmeister
Ralf Reussner

Organization

General Chair

Ivica Crnkovic, Mälardalen University, Sweden

Program Committee Chair

Christine Hofmeister, Lehigh University, USA

Steering Committee

Ralf Reussner, University of Karlsruhe (TH), Germany

Judith Stafford, Tufts University, USA

Sven Overhage, Augsburg University, Germany

Steffen Becker, University of Karlsruhe (TH), Germany

Program Committee

Colin Atkinson, University of Mannheim, Germany

Len Bass, Software Engineering Institute, USA

Don Batory, University of Texas at Austin, USA

PerOlof Bengtsson, University of Karlskrona/Ronneby, Sweden

Jan Bosch, Nokia Research Center, Netherlands

Alexander Brandle, Microsoft Research, UK

Michel Chaudron, Technische Universiteit Eindhoven, Netherlands

Viktoria Firus, University of Oldenburg, Germany

Hassan Gomaa, George Mason University, USA

Ian Gorton, National ICT, Australia

Volker Gruhn, University of Leipzig, Germany

Wilhelm Hasselbring, University of Oldenburg / OFFIS, Germany

Jean-Marc Jezequel, University of Rennes / INRIA, France

Philippe Kruchten, University of British Columbia, Canada

Patricia Lago, Vrije Universiteit, Netherlands

Nicole Levy, University of Versailles, France

Tomi Mannisto, Helsinki University of Technology, Finland

Raffaella Mirandola, Politecnico di Milano, Italy

Robert Nord, Software Engineering Institute, USA

Frantisek Plasil, Charles University, Czech Republic

Iman Poernomo, King's College, UK

Sasikumar Punnekkat, Mälardalen University, Sweden

Andreas Rausch, University of Kaiserslautern, Germany

VIII Organization

Matthias Riebisch, Technical University of Ilmenau, Germany
Bernhard Rumpe, University of Technology Braunschweig, Germany
Chris Salzmänn, BMW Car-IT
Jean-Guy Schneider, Swinburne University, Australia
Michael Stal, Siemens, Germany
Clemens Szyperski, Microsoft, USA
Hans van Vliet, Vrije Universiteit, Netherlands
Wolfgang Weck, Independent Software Architect, Switzerland

Additional Reviewers

Sven Apel, University of Magdeburg, Germany
Roberto Lopez-Herrejon, University of Oxford, UK
Moreno Marzolla, INFN, Italy

Sponsoring Institutions

University of Karlsruhe (TH)
Mälardalen University
Västerås City

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Moshe Y. Vardi

Rice University, Houston, TX, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Lecture Notes in Computer Science

For information about Vols. 1–4233

please contact your bookseller or Springer

Vol. 4337: S. Arun-Kumar, N. Garg (Eds.), *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science*. XIII, 430 pages. 2006.

Vol. 4331: G. Min, B. Di Martino, L.T. Yang, M. Guo, G. Ruenger (Eds.), *Frontiers of High Performance Computing and Networking – ISPA 2006 Workshops*. XXVII, 1141 pages. 2006.

Vol. 4329: R. Barua, T. Lange (Eds.), *Progress in Cryptology – INDOCRYPT 2006*. X, 454 pages. 2006.

Vol. 4326: S. Göbel, R. Malkewitz, I. Iurgel (Eds.), *Technologies for Interactive Digital Storytelling and Entertainment*. X, 384 pages. 2006.

Vol. 4325: J. Cao, I. Stojmenovic, X. Jia, S.K. Das (Eds.), *Mobile Ad-hoc and Sensor Networks*. XIX, 887 pages. 2006.

Vol. 4318: H. Lipmaa, M. Yung, D. Lin (Eds.), *Information Security and Cryptology*. XI, 305 pages. 2006.

Vol. 4313: T. Margaria, B. Steffen (Eds.), *Leveraging Applications of Formal Methods*. IX, 197 pages. 2006.

Vol. 4312: S. Sugimoto, J. Hunter, A. Rauber, A. Morishima (Eds.), *Digital Libraries: Achievements, Challenges and Opportunities*. XVIII, 571 pages. 2006.

Vol. 4311: K. Cho, P. Jacquet (Eds.), *Technologies for Advanced Heterogeneous Networks II*. XI, 253 pages. 2006.

Vol. 4307: P. Ning, S. Qing, N. Li (Eds.), *Information and Communications Security*. XIV, 558 pages. 2006.

Vol. 4306: Y. Avrithis, Y. Kompatsiaris, S. Staab, N.E. O'Connor (Eds.), *Semantic Multimedia*. XII, 241 pages. 2006.

Vol. 4305: A.A. Shvartsman (Ed.), *Principles of Distributed Systems*. XIII, 441 pages. 2006.

Vol. 4304: A. Sattar, B.-H. Kang (Eds.), *AI 2006: Advances in Artificial Intelligence*. XXVII, 1303 pages. 2006. (Sublibrary LNAI).

Vol. 4302: J. Domingo-Ferrer, L. Franconi (Eds.), *Privacy in Statistical Databases*. XI, 383 pages. 2006.

Vol. 4301: D. Pointcheval, Y. Mu, K. Chen (Eds.), *Cryptology and Network Security*. XIII, 381 pages. 2006.

Vol. 4300: Y.Q. Shi (Ed.), *Transactions on Data Hiding and Multimedia Security I*. IX, 139 pages. 2006.

Vol. 4296: M.S. Rhee, B. Lee (Eds.), *Information Security and Cryptology – ICISC*. XIII, 358 pages. 2006.

Vol. 4295: J.D. Carswell, T. Tezuka (Eds.), *Web and Wireless Geographical Information Systems*. XI, 269 pages. 2006.

Vol. 4293: A. Gelbukh, C.A. Reyes-Garcia (Eds.), *MICA 2006: Advances in Artificial Intelligence*. XXVIII, 1232 pages. 2006. (Sublibrary LNAI).

Vol. 4292: G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. Nefian, G. Meenakshisundaram, V. Pascucci, J. Zara, J. Molineros, H. Theisel, T. Malzbender (Eds.), *Advances in Visual Computing, Part II*. XXXII, 906 pages. 2006.

Vol. 4291: G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. Nefian, G. Meenakshisundaram, V. Pascucci, J. Zara, J. Molineros, H. Theisel, T. Malzbender (Eds.), *Advances in Visual Computing, Part I*. XXXI, 916 pages. 2006.

Vol. 4290: M. van Steen, M. Henning (Eds.), *Middleware 2006*. XIII, 425 pages. 2006.

Vol. 4289: M. Ackermann, B. Berendt, M. Grobelnik, A. Hotho, D. Mladenič, G. Semeraro, M. Spiliopoulou, G. Stumme, V. Svatek, M. van Someren (Eds.), *Semantics, Web and Mining*. X, 197 pages. 2006. (Sublibrary LNAI).

Vol. 4288: T. Asano (Ed.), *Algorithms and Computation*. XX, 766 pages. 2006.

Vol. 4285: Y. Matsumoto, R. Sproat, K.-F. Wong, M. Zhang (Eds.), *Computer Processing of Oriental Languages*. XVII, 544 pages. 2006. (Sublibrary LNAI).

Vol. 4284: X. Lai, K. Chen (Eds.), *Advances in Cryptology – ASIACRYPT 2006*. XIV, 468 pages. 2006.

Vol. 4283: Y.Q. Shi, B. Jeon (Eds.), *Digital Watermarking*. XII, 474 pages. 2006.

Vol. 4282: Z. Pan, A. Cheok, M. Haller, R.W.H. Lau, H. Saito, R. Liang (Eds.), *Advances in Artificial Reality and Tele-Existence*. XXIII, 1347 pages. 2006.

Vol. 4281: K. Barkaoui, A. Cavalcanti, A. Cerone (Eds.), *Theoretical Aspects of Computing – ICTAC 2006*. XV, 371 pages. 2006.

Vol. 4280: A.K. Datta, M. Gradinariu (Eds.), *Stabilization, Safety, and Security of Distributed Systems*. XVII, 590 pages. 2006.

Vol. 4279: N. Kobayashi (Ed.), *Programming Languages and Systems*. XI, 423 pages. 2006.

Vol. 4278: R. Meersman, Z. Tari, P. Herrero (Eds.), *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Part II*. XLV, 1004 pages. 2006.

Vol. 4277: R. Meersman, Z. Tari, P. Herrero (Eds.), *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Part I*. XLV, 1009 pages. 2006.

Vol. 4276: R. Meersman, Z. Tari (Eds.), *On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE, Part II*. XXXII, 752 pages. 2006.

Vol. 4275: R. Meersman, Z. Tari (Eds.), *On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE, Part I*. XXXI, 1115 pages. 2006.

- Vol. 4274: Q. Huo, B. Ma, E.-S. Chng, H. Li (Eds.), Chinese Spoken Language Processing. XXIV, 805 pages. 2006. (Sublibrary LNAI).
- Vol. 4273: I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, L. Aroyo (Eds.), The Semantic Web - ISWC 2006. XXIV, 1001 pages. 2006.
- Vol. 4272: P. Havinga, M. Lijding, N. Meratnia, M. Wegdam (Eds.), Smart Sensing and Context. XI, 267 pages. 2006.
- Vol. 4271: F.V. Fomin (Ed.), Graph-Theoretic Concepts in Computer Science. XIII, 358 pages. 2006.
- Vol. 4270: H. Zha, Z. Pan, H. Thwaites, A.C. Addison, M. Forte (Eds.), Interactive Technologies and Sociotechnical Systems. XVI, 547 pages. 2006.
- Vol. 4269: R. Sate, S. van der Meer, D. O'Sullivan, T. Pfeifer (Eds.), Large Scale Management of Distributed Systems. XIII, 282 pages. 2006.
- Vol. 4268: G. Parr, D. Malone, M. Ó Foghlú (Eds.), Autonomic Principles of IP Operations and Management. XIII, 237 pages. 2006.
- Vol. 4267: A. Helmy, B. Jennings, L. Murphy, T. Pfeifer (Eds.), Autonomic Management of Mobile Multimedia Services. XIII, 257 pages. 2006.
- Vol. 4266: H. Yoshiura, K. Sakurai, K. Rannenber, Y. Murayama, S. Kawamura (Eds.), Advances in Information and Computer Security. XIII, 438 pages. 2006.
- Vol. 4265: L. Todorovski, N. Lavrač, K.P. Jantke (Eds.), Discovery Science. XIV, 384 pages. 2006. (Sublibrary LNAI).
- Vol. 4264: J.L. Balcázar, P.M. Long, F. Stephan (Eds.), Algorithmic Learning Theory. XIII, 393 pages. 2006. (Sublibrary LNAI).
- Vol. 4263: A. Levi, E. Savaş, H. Yenigün, S. Balcişoy, Y. Saygın (Eds.), Computer and Information Sciences - ISCIS 2006. XXIII, 1084 pages. 2006.
- Vol. 4262: K. Havelund, M. Núñez, B. Wolff, G. Roşu (Eds.), Formal Approaches to Software Testing and Runtime Verification. VIII, 255 pages. 2006.
- Vol. 4261: Y. Zhuang, S. Yang, Y. Rui, Q. He (Eds.), Advances in Multimedia Information Processing - PCM 2006. XXII, 1040 pages. 2006.
- Vol. 4260: Z. Liu, J. He (Eds.), Formal Methods and Software Engineering. XII, 778 pages. 2006.
- Vol. 4259: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H.S. Nguyen, R. Słowiński (Eds.), Rough Sets and Current Trends in Computing. XXII, 951 pages. 2006. (Sublibrary LNAI).
- Vol. 4257: I. Richardson, P. Runeson, R. Messnarz (Eds.), Software Process Improvement. XI, 219 pages. 2006.
- Vol. 4256: L. Feng, G. Wang, C. Zeng, R. Huang (Eds.), Web Information Systems - WISE 2006 Workshops. XIV, 320 pages. 2006.
- Vol. 4255: K. Aberer, Z. Peng, E.A. Rundensteiner, Y. Zhang, X. Li (Eds.), Web Information Systems - WISE 2006. XIV, 563 pages. 2006.
- Vol. 4254: T. Grust, H. Höpfner, A. Illarramendi, S. Jablonski, M. Mesiti, S. Müller, P.-L. Patranjan, K.-U. Sattler, M. Spiliopoulou, J. Wijsen (Eds.), Current Trends in Database Technology - EDBT 2006. XXXI, 932 pages. 2006.
- Vol. 4253: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. XXXII, 1301 pages. 2006. (Sublibrary LNAI).
- Vol. 4252: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. XXXIII, 1335 pages. 2006. (Sublibrary LNAI).
- Vol. 4251: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXVI, 1297 pages. 2006. (Sublibrary LNAI).
- Vol. 4250: H.J. van den Herik, S.-C. Hsu, T.-s. Hsu, H.H.L.M. Donkers (Eds.), Advances in Computer Games. XIV, 273 pages. 2006.
- Vol. 4249: L. Goubin, M. Matsui (Eds.), Cryptographic Hardware and Embedded Systems - CHES 2006. XII, 462 pages. 2006.
- Vol. 4248: S. Staab, V. Svátek (Eds.), Managing Knowledge in a World of Networks. XIV, 400 pages. 2006. (Sublibrary LNAI).
- Vol. 4247: T.-D. Wang, X. Li, S.-H. Chen, X. Wang, H. Abbass, H. Iba, G. Chen, X. Yao (Eds.), Simulated Evolution and Learning. XXI, 940 pages. 2006.
- Vol. 4246: M. Hermann, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 588 pages. 2006. (Sublibrary LNAI).
- Vol. 4245: A. Kuba, L.G. Nyúl, K. Palágyi (Eds.), Discrete Geometry for Computer Imagery. XIII, 688 pages. 2006.
- Vol. 4244: S. Spaccapietra (Ed.), Journal on Data Semantics VII. XI, 267 pages. 2006.
- Vol. 4243: T. Yakhno, E.J. Neuhold (Eds.), Advances in Information Systems. XIII, 420 pages. 2006.
- Vol. 4242: A. Rashid, M. Aksit (Eds.), Transactions on Aspect-Oriented Software Development II. IX, 289 pages. 2006.
- Vol. 4241: R.R. Beichel, M. Sonka (Eds.), Computer Vision Approaches to Medical Image Analysis. XI, 262 pages. 2006.
- Vol. 4239: H.Y. Youn, M. Kim, H. Morikawa (Eds.), Ubiquitous Computing Systems. XVI, 548 pages. 2006.
- Vol. 4238: Y.-T. Kim, M. Takano (Eds.), Management of Convergence Networks and Services. XVIII, 605 pages. 2006.
- Vol. 4237: H. Leitold, E. Markatos (Eds.), Communications and Multimedia Security. XII, 253 pages. 2006.
- Vol. 4236: L. Breveglieri, I. Koren, D. Naccache, J.-P. Seifert (Eds.), Fault Diagnosis and Tolerance in Cryptography. XIII, 253 pages. 2006.
- Vol. 4234: I. King, J. Wang, L. Chan, D. Wang (Eds.), Neural Information Processing, Part III. XXII, 1227 pages. 2006.

¥359.00元

Table of Contents

Abstracts of the Keynotes

Expanding the Scope of Software Product Families: Problems and Alternative Approaches	1
<i>Jan Bosch</i>	
Composing with Style – Components Meet Architecture	2
<i>Clemens Szyperski</i>	

Abstracts of the Tutorials

Documentation Principles and Practices that You Can Live With	3
<i>Judith Stafford</i>	
Model-Based Software Development with Eclipse	5
<i>Ralf Reussner, Steffen Becker</i>	
Software Architecture Analysis and Evaluation	7
<i>Heinz Züllighoven, Carola Lilienthal, Marcel Benniscke</i>	

Architecture Evaluation: Selecting Alternatives

MEMS: A Method for Evaluating Middleware Architectures	9
<i>Yan Liu, Ian Gorton, Len Bass, Cuong Hoang, Suhail Abanmi</i>	
Evaluating Alternative COTS Assemblies from Imperfect Component Information	27
<i>Hernán Astudillo, Javier Pereira, Claudia López</i>	

Managing and Applying Architectural Knowledge

Building Up and Reasoning About Architectural Knowledge	43
<i>Philippe Kruchten, Patricia Lago, Hans van Vliet</i>	
Managing Architectural Design Decisions for Safety-Critical Software Systems	59
<i>Weihang Wu, Tim Kelly</i>	

Architectural Evaluation: Performance Prediction

Runtime Prediction of Queued Behaviour	78
<i>Nurzhan Duzbayev, Iman Poernomo</i>	

Model Transformation in Software Performance Engineering 95
Antinisca Di Marco, Raffaella Mirandola

Processes for Supporting Architecture Quality

Traveling Architects – A New Way of Herding Cats 111
Aino Vonge Corry, Klaus Marius Hansen, David Svensson

A Practical Architecture-Centric Analysis Process 127
Antonio Bucchiarone, Henry Muccini, Patrizio Pelliccione

Models for Architecture Evaluation

Embedded Systems Architecture: Evaluation and Analysis 145
Bastian Florentz, Michaela Huhn

Parameter Dependent Performance Specifications of Software
Components 163
Heiko Koziol, Jens Happe, Steffen Becker

Architectural Evaluation

Applying the ATAM to an Architecture for Decentralized Control
of a Transportation System 180
Nelis Boucké, Danny Weyns, Kurt Schelfhout, Tom Holvoet

Towards an Integration of Standard Component-Based Safety
Evaluation Techniques with SaveCCM 199
Lars Grunske

Author Index 215

Expanding the Scope of Software Product Families: Problems and Alternative Approaches

Jan Bosch

Software and Application Technologies Laboratory
Nokia Research Center, Helsinki, Finland

Abstract. Software product families have found broad adoption in the embedded systems industry. Product family thinking has been prevalent in this context for mechanics and hardware and adopting the same for software has been viewed as a logical approach. During recent years, however, the trends of convergence, end-to-end solutions, shortened innovation and R&D cycles and differentiation through software engineering capabilities have lead to a development where organizations are stretching the scope of their product families far beyond the initial design. Failing to adjust the product family approach, including the architectural and process dimensions when the business strategy is changing is leading to several challenging problems that can be viewed as symptoms of this approach. The keynote discusses the key symptoms, the underlying causes for these symptoms as well as solutions for realigning the product family approach with the business strategy.

About

Prof. dr. ir. **Jan Bosch** is a Vice President and head of the Software and Application Technologies Laboratory at Nokia Research Center, Finland. Earlier, he headed the software engineering research group at the University of Groningen, The Netherlands, where he holds a professorship in software engineering. He received a MSc degree from the University of Twente, The Netherlands, and a PhD degree from Lund University, Sweden. His research activities include software architecture design, software product families, software variability management and component-oriented programming. He is the author of a book “Design and Use of Software Architectures: Adopting and Evolving a Product Line Approach” published by Pearson Education (Addison-Wesley & ACM Press), (co-)editor of several books and volumes in, among others, the Springer LNCS series and (co-)author of a significant number of research articles. He has been guest editor for journal issues, chaired several conferences as general and program chair, served on many program committees and organized numerous workshops. Finally, he is and has been a member of the steering groups of the GCSE and WICSA conferences.

Composing with Style – Components Meet Architecture

(Invited Talk)

Clemens Szyperski

Microsoft Research
One Microsoft Way, Redmond WA 98052, USA

Abstract. Composability itself is probably the least composable term in the theory of computer science. In this talk, I'll explore some of the troubling reasons why we have succeeded only so-so when it comes to the creation of composable software - and thus software components. Architecture can often come to the rescue, but only when applied with great style.

About

Clemens Szyperski joined Microsoft Research as a Software Architect in 1999. His team moved into a product incubation phase in 2001 and began production development in early 2003. A first product developed in an entirely new way will be released together with the upcoming Office System 2007. Since late 2005 he is now working on driving novel platform technology in Microsoft's new Connected Systems Division. His focus is on the end-to-end issues of leveraging component software to effectively build new kinds of software. He maintains an affiliation with Microsoft Research and continues his general activities in the wider research arena. His Jolt-award-winning book *Component Software* (Addison Wesley) appeared in a fully revised and extended second edition in late 2002. *Software Ecosystem* (MIT Press), co-authored with Dave Messerschmitt of UC Berkeley, was published in mid 2003. Clemens serves on numerous program committees, including ECOOP, ESEC/FSE, ICSE, and OOPSLA. He served as assessor and panelist for national funding bodies in Australia, Canada, Ireland, the Netherlands, and USA. He is a cofounder of Oberon microsystems, Zurich, Switzerland, and its now-public spin-off esmertec.

From 1994 to 1999, he was an associate professor at the School of Computer Science, Queensland University of Technology, Australia, where he retains an adjunct professorship. He held a postdoc scholarship at ICSI, affiliated with UC Berkeley in 1992/93. In 1992, he received his PhD in computer science from the Swiss Federal Institute of Technology (ETH) in Zurich under Prof. Niklaus Wirth and in 1987 his Masters in electrical engineering/ computer engineering from Aachen University of Technology (RWTH).

Documentation Principles and Practices That You Can Live with

Judith Stafford

Tufts University, Boston, USA

Abstract. Software architecture has become a widely-accepted conceptual basis for the development of non-trivial software in all application areas and by organizations of all sizes. Effectively documenting an architecture is as important as crafting it; if the architecture is not understood, or worse, misunderstood, it cannot meet its goals as the unifying vision for software development. Development-based architecture strategies, such as Rational's Unified Process, stop short of prescribing documentation standards. The Views and Beyond approach to software architecture provides practical guidance on the what, why, and how of creating IEEE 1471-2000 compliant documentation for your software architecture that will be used for years to come. The approach is based on the well-known concept of views and is presented in the context of prevailing prescriptive models for architecture, including the Unified Process and UML 2.0, which has improved support for representing key architectural elements over its predecessors.

Attendee Background

Participants should have experience with creating or using descriptions of large software systems and some knowledge of the Unified Modeling Language.

Tutorial Objectives

The primary aim of this tutorial is to teach developers what constitutes good documentation of a software architecture, why it is worth the effort to create and maintain a documentation package, and how to write it down. A secondary aim is to teach other stakeholders why they should care about architectural documentation and how they can use it to make their life easier, increase productivity, and decrease overall system development and maintenance costs.

About

Judith Stafford is a Senior Lecturer in the Department of Computer Science at Tufts University, and is also a visiting scientist at the Software Engineering Institute, Carnegie Mellon University. Dr. Stafford has worked for several years

in the area of compositional reasoning and its application to software architectures and component-based systems. She has organized workshops, given invited talks, taught tutorials, and written widely in these areas including co-authoring the book that inspired this tutorial, *Documenting Software Architectures: Views and Beyond*, Addison Wesley, 2002 and several book chapters on software architecture and component-based software engineering.

Model-Based Software Development with Eclipse

Ralf Reussner and Steffen Becker

University of Karlsruhe (TH), Germany

Abstract. The tutorial consists of two parts. In the first part (45 min), Ralf Reussner focuses on the importance of an explicitly modelled software architecture. Besides an introduction into common architectural views, the role of the software architect is compared to "classical building" architects. As part of this, the often used comparison between building architecture and software architecture is critically reviewed. In particular, the role of an architect is discussed in model-driven software projects.

During the second part of the tutorial (135 min), Steffen Becker demonstrates online model driven development tools based on Eclipse. First, an introduction is given on the metamodeling tools of the Eclipse Modelling Framework (EMF) and on the Graphical Modelling Framework (GMF) used to generate a domain specific editors for user defined (meta-)models. Additionally, the MDA framework of the OMG is presented and the concepts are applied to the introduced tools.

A live demonstration of the capabilities of the introduced tools for model transformations shows finally how a domain specific modelling tool can be generated to a large extend automatically using an EMF-model instance and the generator of GMF. As a result, an editor based on the Eclipse Graphical Editing Framework (GEF) can be deployed and run using Eclipse.

About

Professor **Ralf Reussner** holds the Chair for Software-Design and -Quality at the University of Karlsruhe since 2006. His research group is well established in the area of component based software design, software architecture and predictable software quality. Professor Reussner shaped this field not only by over 60 peer-reviewed publications in Journals and Conferences, but also by establishing various conferences and workshops. In addition, he acts as a PC member or reviewer of several conferences and journals. As Director of Software Engineering at the Informatics Research Centre in Karlsruhe (FZI) he consults various industrial partners in the areas of component based software, architectures and software quality. He is principal investigator or chief coordinator in several grants from industrial and governmental funding agencies. He graduated from University of Karlsruhe with a PhD in 2001. After this, Ralf was a Senior Research Scientist and project-leader at the Distributed Systems Technology Centre (DSTC Pty Ltd), Melbourne, Australia. From March 2003 till January