Lecture Notes in Computer Science 5525

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Betty H.C. Cheng   Rogério de Lemos
Holger Giese   Paola Inverardi   Jeff Magee (Eds.)

Software Engineering for Self-Adaptive Systems

Springer

Volume Editors

Betty H.C. Cheng
Michigan State University, Department of Computer Science and Engineering
3115 Engineering Building, East Lansing, MI 48824-1226, USA
E-mail: chengb@cse.msu.edu

Rogério de Lemos
University of Kent, Computing Laboratory
Canterbury, Kent CT2 7NF, UK
E-mail: r.delemos@kent.ac.uk

Holger Giese
Hasso Plattner Institute for Software Systems Engineering
Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
E-mail: holger.giese@hpi.uni-potsdam.de

Paola Inverardi
Università dell'Aquila, Dipartimento di Informatica
67100 L'Aquila, Italy
E-mail: inverard@di.univaq.it

Jeff Magee
Imperial College, Department of Computing
180 Queen's Gate, London SW7 2BZ, UK
E-mail: j.magee@imperial.ac.uk

Library of Congress Control Number: 2009928525

CR Subject Classification (1998): D.2, D.3, F.1.1, I.2.2

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-642-02160-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-02160-2 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12686551 06/3180 5 4 3 2 1 0

Preface

The complexity of current software-based systems has led the software engineering community to look for inspiration in diverse related fields (e.g., robotics and control theory) as well as other areas (e.g., biology) to find innovative approaches for building, running, and managing software systems and services. Therefore, *self-adaptation* – systems that are able to adjust their behavior at run-time in response to their perception of the environment and the system itself – has become a hot topic within the software engineering community.

This book and the roadmap paper that is included here are two key outcomes from the Dagstuhl Seminar 08031 on "Software Engineering for Self-Adaptive Systems" that took place in January 2008. In addition to the roadmap paper, this book includes invited papers from recognized experts in the area that describe the current state of the art in the field, and papers that provide an insight into the key features of self-adaptive systems and how these should be designed. All the papers were peer-reviewed, with the exception of the roadmap paper, which was based on the discussion held at the Dagstuhl Seminar and put together by several of its participants. The book consists of four parts: "Research Roadmap," "Architecture-Based Self-Adaptation," "Context-Aware and Model-Driven Self-Adaptation," and "Self-Healing."

The first part entitled "Research Roadmap" includes the roadmap paper on research challenges for the area of software engineering for self-adaptive systems as well as the two papers that are extended versions of two self-adaptation views that were presented in the roadmap paper.

With the contribution of several authors—participants of the Dagstuhl Seminar on Software Engineering for Self-Adaptive Systems—the roadmap paper, which is entitled "Software Engineering for Self-Adaptive Systems: A Research Roadmap" summarizes the state of the art and identifies critical challenges for the systematic software engineering of self-adaptive systems.

The second paper of this part, authored by J. Andersson, R. de Lemos, S. Malek and D. Weyns, entitled "Modelling Dimensions for Self-Adaptive Systems" proposes a classification of modelling dimensions for self-adaptive software systems, which are key aspects that characterize self-adaptation. The identified modelling dimensions are illustrated by applying them to several application scenarios.

The third paper "Engineering Self-Adaptive Systems Through Feedback Loops," which is authored by Y. Brun, G. Di Marzo Serugendo, C. Gacek, H. Giese, H. Kienle, M. Litoiu, H. Muller, M. Pezzè and M. Shaw, promotes, in the design of self-adaptive systems, the use of feedback loops as first-class entities. The paper argues that feedback loops are essential for understanding all types of self-adaptive systems, and identifies critical challenges that must be addressed to enable systematic and well-organized engineering of self-adaptive and self-managing software systems.

Part two of the book entitled "Architecture-Based Self-Adaptation" consists of three papers describing solutions in which architectures take a central role in the development of self-adaptive software systems.

The first paper by S.-W. Cheng, V. V. Poladian, D. Garlan and B. Schmerl, entitled "Improving Architecture-Based Self-Adaptation Through Resource Prediction," discusses how self-adaptation using architectural models can be improved by adopting an anticipatory approach in which predictions are used to inform adaptation strategies rather than operate in a purely reactive way. It is demonstrated that predictions can be incorporated into an architecture-based adaptation framework showing the resulting benefits.

J.C. Georgas and R.N. Taylor in the paper "Policy-Based Architectural Adaptation Management: Robotics Domain Case Studies" continue previous work in which the authors developed notations and tools that support the design and development of policy- and architecture-based self-adaptive systems that are modular and have the ability to change the specifications of the adaptation policy during system run-time. This paper assesses the feasibility of integrating those notations and tools with the robotics domain, develops novel self-adaptive capabilities for robotic systems, and identifies difficulties for such integration.

The last paper of this part, authored by W. Heaven, D. Sykes, J. Magee and J. Kramer and entitled "A Case Study in Goal-Driven Architectural Adaptation," presents an approach to constructing autonomous systems that synthesize tasks from high-level goals. In order to execute these tasks reliably in a changing environment, the software architecture of the system is adapted. The applicability of the proposed approach is demonstrated with a case study involving mobile robots.

Part three of the book is "Context-Aware and Model-Driven Self-Adaptation," and includes five papers centered around context-aware self-adaptability and how model-driven approaches can be applied in the development of self-adaptive software systems.

O. Nierstrasz, M. Denker and L. Renggli are the authors of "Model-Centric, Context-Aware Software Adaptation," the first paper of this part. This paper makes the case for model-centric and context-aware software adaptation, and shows through concrete examples how these design principles work at the level of application interface, programming language and run-time. The paper also discusses how the presence of sufficiently high-level models at run-time can enable very dynamic forms of context-dependent software adaptation. The authors also present a research agenda for model-centric development that supports dynamic software adaptation and evolution.

The second paper, entitled "Modeling of Context-Aware Self-Adaptive Applications in Ubiquitous and Service-Oriented Environments" written by K. Geihs, R. Reichle, M. Wagner and M. Ullah Khan, presents a modelling approach for the integration of service-based adaptation in a planning framework for compositional adaptation of context-aware applications. Based on semantic descriptions that are associated to variation points in the component framework, the proposed modelling framework provides a harmonized view on QoS-properties of external discoverable services and conventional context properties of component-based applications.

The third paper "MUSIC: Middleware Support for Self-Adaptation in Ubiquitous and Service-Oriented Environments," authored by R. Rouvoy, P. Barone, Y. Ding, F. Eliassen, S. Hallsteinsen, J. Lorenzo, A. Mamelli and U. Scholz, introduces an extension of the MUSIC component-based planning framework. This extension optimizes the overall utility of applications by plug-in interchangeable components and services when there are unexpected changes of the service provider landscape in a ubiquitous context. The planning framework is outlined and a motivating scenario is presented.

Paper four of this part, "Using Architecture Models to Support the Generation and Operation of Component-Based Adaptive Systems," authored by N. Bencomo and G. Blair, presents an approach that uses architectural models for supporting the generation and operation of component-based adaptive systems. The proposed approach supports the specification and validation of models based on abstractions of architectural or structural variability as well as environment and context variability.

Finally, the paper written by V. Grassi, R. Mirandola and E. Randazzo, and entitled "Model-Driven Assessment of QoS-Aware Self-Adaptation," presents an approach for supporting the QoS assessment of self-adaptable systems, which extends an intermediate modelling language for capturing the core features of a dynamically adaptable architecture model from a performance/dependability viewpoint. A model transformation chain is outlined that maps a "design oriented" model to an "analysis oriented" model for permitting the application of a suitable analysis methodology.

Part four of the book is "Self-Healing," and contains two papers related to error detection and system recovery.

The first paper in this part is authored by M. Pezzè and J. Wuttke with the title "Automatic Generation of Runtime Failure Detectors from Property Templates." The authors propose property templates to link requirements and design, and to generate automatically assertions using a model-based specification language. It targets self-healing software and permits detecting failures at low-cost at run-time while providing high detection precision and enough information about the detected failures to enable automatic healing actions.

The final paper of this part "Filtered Cartesian Flattening and Microrebooting to Build Enterprise Applications with Self-adaptive Healing," written by J. White, B. Dougherty, H.D. Strowd and D.C. Schmidt, considers the development of enterprise applications that can self-adapt to tolerate component failures. The paper describes a microrebooting technique based on feature models and a heuristic algorithm for deriving new configurations, as well as a container component that crashes the faulty subsystem and reboots the new configuration. For feature selection, the approach uses Filtered Cartesian Flattening and multidimensional multiple-choice knapsack heuristic algorithms in order to reduce self-healing time.

Although the self-adaptability of systems has been studied in a wide range of disciplines, from biology to robotics, only recently has the software engineering community recognized its key role in enabling the development of future software systems that are able to self-adapt to changes that may occur in the system, its requirements, or the environment in which it is deployed. In our understanding, this volume is one of the first books containing a collection of papers that looks specifically into the current state of the art in the field, describes a wide range of approaches coming from different strands of software engineering, and presents future challenges facing this always resurgent and challenging field of research. We are certain that this book will prove valuable both for practitioners and researchers that are involved with the development of self-adaptive software systems.

Our thanks go to the authors of the contributions for their excellent work, the participants of the Dagstuhl Seminar on Software Engineering for Self-Adaptive Systems for their active participation in the discussions and further contributions, and Alfred Hofmann and his team from Springer for believing in this important topic and for helping us to get the book published. Last but not least, we highly appreciate the efforts of our reviewers who have helped us in ensuring the high quality of the contributions. They are J. Andersson, L. Baresi, N. Bencomo, G. Blair, J. Bradbury, Y. Brun, C. Canal, G. Candea, S.-W. Cheng, C. Cuesta, G. Di Marzo Serugendo, Y. Ding, S. Dustdar, C. Gacek, K. Geihs, J. Georgas, K. M. Goeschka, V. Grassi, R. Hirschfeld, J. Kramer, E. Letier, M. Litoiu, J. Liu, P. Lollini, S. Malek, J. A. McCann, R. Mirandola, H. Müller, J. Noyé, O. Nierstrasz, P. Popov, P. Robertson, M. Sadjadi, G. Salaün, B. Schmerl, U. Scholz, J. P. Sousa, R. Taylor, M. Tichy, M. Tivoli, D. Weyns, and several anonymous reviewers.

March 2009

Betty H.C. Cheng
Rogério de Lemos
Holger Giese
Paola Inverardi
Jeff Magee

Table of Contents

Part 4: Self-Healing