

Self-Aware Computing Systems

Samuel Kounev · Jeffrey O. Kephart
Aleksandar Milenkoski · Xiaoyun Zhu
Editors

Self-Aware Computing Systems

Editors

Samuel Kounev
University of Würzburg
Würzburg
Germany

Xiaoyun Zhu
VMWare Inc.
Santa Clara, CA
USA

Jeffrey O. Kephart
Thomas J. Watson Research Center
Hawthorne, NY
USA

and

Futurewei Technologies, Inc.
Santa Clara, CA
USA

Aleksandar Milenkoski
University of Würzburg
Würzburg
Germany

ISBN 978-3-319-47472-4

ISBN 978-3-319-47474-8 (eBook)

DOI 10.1007/978-3-319-47474-8

Library of Congress Control Number: 2016954693

© Springer International Publishing AG 2017

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.

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.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Background

During the past several years, many different research communities have explored the aspects of self-awareness in computing systems, each from their own perspective. To the artificial intelligence community, the natural unit of self-awareness is the software agent; to those who study the autonomic computing, it is the autonomic element. One can identify at least a dozen other research communities for which the *self-awareness* of a computing system is a central issue. However, the underlying commonalities in these notions of self-awareness are often obscured by the differences in the nomenclature and the lack of precise definitions.

Recently, there have been efforts to put a spotlight on the topic of self-awareness in the computing systems and treat it as a worthwhile object of study in its own right. Examples include the SELF-aware Computing (SEEC)¹ project at MIT and University of Chicago, the ASCENS² and EPiCS³ FP7 EU Projects, and the FOCAS FET Coordination Action.⁴ There have also been efforts to assemble the researchers with a common interest in self-adaptive systems and to publish compendia of works on the topic—a notable example being the SEAMS Dagstuhl seminars and workshop series. However, such collections have tended to group contributions either by community or by theme, leaving the reader without a single clear unifying definition of self-aware computing that applies broadly across multiple domains.

In an effort to establish a clear definition and understanding of self-aware computing, we organized a Dagstuhl seminar⁵ entitled “Model-driven Algorithms and Architectures for Self-Aware Computing Systems,” which was held in the week

¹<http://hdl.handle.net/1721.1/67020>.

²<http://www.ascens-ist.eu/>.

³http://cordis.europa.eu/project/rcn/95042_en.html.

⁴<http://www.focas.eu/>.

⁵<http://www.dagstuhl.de/15041>.

of January 18–23, 2015. The seminar brought together researchers from several different communities to discuss the past and future trends in self-aware computing and encourage the active collaborations and cross-fertilization among related research efforts. Special attention was paid to achieve the diversity along several dimensions, including the field of study, age, and gender, with stronger-than-usual participation from industry. Over 40 participants presented brief perspectives on self-awareness from their respective fields, interleaved with long periods of discussion aimed at developing a cross-disciplinary understanding of self-awareness. A key outcome from one such discussion was a consensus definition of the term “self-aware computing system” that was both broad enough to cover the interdisciplinary research landscape and precise enough to distinguish the aspects of those disciplines that are relevant to self-awareness from those that are not. Roughly speaking, we say that computing systems are self-aware if they possess the capability to learn and exploit the models of themselves and the environment in which they are situated so as to act in accordance with the high-level goals.

Seeking to capitalize on the considerable positive energy and momentum created by the seminar, many of the participants agreed to contribute to a book on self-aware computing systems. Rather than following the usual formula of organizing the seminar presentations into a book, we decided to write the book completely from scratch and to organize the writing effort in such a way that the chapters would be well-integrated and cross-referenced, and based upon the collaborative efforts and perspectives of authors from multiple disciplines.

Given this ambition and approach, writing this book proved to be an enormous undertaking, and one that overachieved on our desire to foster fruitful and ongoing collaborations among researchers who might never have worked together had they not met at Dagstuhl. Discussions regarding the purpose and structure of this book began during the seminar and continued online for several weeks afterward. Once we reached a tentative agreement about which chapters should be written, and by whom, the authors of each chapter convened numerous times, working individually and collectively across many institutions (both academic and industrial) and time zones to write the initial drafts. About six months after the Dagstuhl seminar, a checkpoint workshop was held at the International Conference on Autonomic Computing 2015 in Grenoble to review the progress of the book and to make some midcourse corrections regarding the content and organization. Some additional authors were recruited to fill the perceived gaps in content and expertise. During the ensuing months, drafts of each chapter were circulated and reviewed to ensure the quality and coherency across chapters (i.e., adequate cross-referencing).

This book is the result of those efforts. Here, professionals, researchers, lecturers, and students will find formal and informal definitions and taxonomies of self-aware computing systems, explanations of how self-aware computing relates to many existing subfields of computer science and software engineering, descriptions of architectures and algorithms for self-aware systems, benefits and pitfalls of self-awareness, reviews of many of the latest relevant research efforts across a wide array of disciplines, and a set of open research challenges. Our hope is that this book will help establish the self-aware computing as a worthy subject of study in its

own right, and inspire the research and development on self-aware computing systems for years to come.

Content

The chapters of this book are organized into five parts: Introduction, System Architectures, Methods and Algorithms, Applications and Case Studies, and Outlook—each of which is now described in further detail.

The four chapters of Part I constitute an introduction that defines *self-aware computing systems* from multiple perspectives and establishes a formal definition, a taxonomy, and a set of reference scenarios that help to unify the remaining chapters of this book. First, in Chap. 1, we propose a formal definition of self-aware computing systems expressed in terms of models, learning, and reasoning. This definition represents a carefully considered consensus among experts from multiple diverse fields who were brought together for the Dagstuhl seminar. Not only does it serve as a motif for the other chapters of this book, but also we hope that the research community will find in this definition a useful and unifying distillation of ideas that contribute to the notion of self-aware computing as a field of study in its own right. The formal definition of self-aware computing systems is supplemented by some positioning of self-aware computing relative to other fields. Chapter 2 elaborates this theme further, relating in greater detail how self-aware computing systems have been described in a dozen fields of study, including control theory, artificial intelligence, autonomic computing, organic computing, and cloud computing. Examining the similarities and the contrasts among these multiple perspectives on self-awareness not only improves one's understanding of the formal definition, but also helps to justify the need for one in the first place. Chapter 3 supplements the formal definition of self-aware computing systems with a theoretical framework that serves as a taxonomy for self-aware computing systems—another motif that recurs frequently throughout the remainder of this book. Finally, Chap. 4 introduces a set of reference scenarios that illustrate how self-aware computing systems differ from their non-self-aware counterparts. The scenarios, which are used and built upon in subsequent chapters of this book, have been chosen to cover a broad range of application areas and to span to the fullest possible extent the taxonomy established in Chap. 3. The simplest, least complex point on this scale is a self-aware sorting algorithm, which nonetheless affords some interesting explorations of self-awareness at design time and run time. The self-aware data center scenario allows us to explore multiple interacting self-aware components representing multiple interests, and several medium- to large-scale cyber-physical systems' scenarios (including smart homes, smart grids, and smart transportation systems) allow us to explore the issues of cooperation, competition, and heterogeneity.

Part II of this book consists of four chapters that explore the architectures for self-aware computing systems. First, Chap. 5 introduces generic concepts and

notations that allow one to describe and compare a wide range of self-aware system architectures in terms of elements and relationships among those elements. Based upon UML (Unified Modeling Language), these architectural concepts follow explicitly from the self-aware computing definition and taxonomy introduced in Part I of this book. Moreover, the smart home reference scenario introduced in Chap. 4 serves as a running illustration of how the notation can be used to describe the architecture of a self-aware system. Subsequent chapters in Part II build upon the foundation laid in Chap. 5. Specifically, Chap. 6 delves into architectures for individual (isolated) self-aware systems, while Chap. 7 treats architectures for collectives in which multiple self-aware systems interact with one another. Both of these chapters explore several points in the self-aware systems taxonomy established in Chap. 3. In Chap. 6, special consideration is given to pre-reflective, reflective, and meta-reflective self-aware system architectures, while in Chap. 7, the major emphasis is placed upon the various types of relationships and organisational patterns that may exist among the interacting self-aware systems that make up the collective, as well as the relationship between the individual and collective self-awareness. The final chapter of Part II, Chap. 8, reviews the present state of reference architectures, architectural frameworks, and languages for self-aware systems and compares them with the ideas presented in the previous chapters of Part II. It also lays out a set of open challenges for self-aware system architectures.

Part III contains seven chapters that focus on the methods and algorithms for self-aware computing systems. The first three chapters treat issues pertaining to system design, such as modeling, synthesis, and verification. First, Chap. 9 discusses the nature, origin, scope, and purpose of self-modeling and its critical role in supporting system self-awareness. An interesting insight is that while models of a system may certainly be imbedded within that system at design time, a system can be much more flexible and adaptive if it possesses the capability to create and modify models of itself at run time—and endowing systems with the ability to learn both the structure and the parameters of such models is an interesting design challenge in itself. Chapter 10 offers a set of strategies for retrofitting self-awareness into legacy non-self-aware systems, and Chap. 11 discusses the synthesis and verification of self-aware computing systems. The next two chapters of Part III discuss a key run-time issue: adaptation. Chapter 12 explores how adaptation arises from an interplay among learning, reasoning, and acting in individual self-aware systems, while Chap. 13 explores how collective interactions among adaptive self-aware entities may lead to disastrous or beneficial emergent behaviors, and offers a number of possible diagnoses and remedies for undesirable behavior. The final two chapters of Part III discuss a variety of approaches for characterizing and measuring the self-awareness. Chapter 14 defines the metrics and benchmarks for the self-aware computing systems, while Chap. 15 discusses the problem of assessing self-awareness from a more general and philosophical perspective.

Part IV contains ten chapters on the applications and case studies in various domains, and the degree to which self-aware computing approaches have been adopted within those domains. The first seven chapters focus on the aspects of self-awareness in the context of cloud computing. Chapters 16 and 17 discuss using

and learning models to support the performance management in data centers, while Chap. 18 treats online workload forecasting, which can be viewed as learning models of the environment in which the performance management system is situated. Chapter 19 reviews the state of virtual machine (VM) management in data centers, which (as also explained in the data center reference scenario section of Chap. 4) falls short of self-awareness for several reasons. One reason is that VM management systems tend to address lower-level metrics rather than adopting the application perspective—a shortcoming that is addressed in Chap. 20, which explores self-awareness of cloud applications. This is followed by two chapters that explore the self-awareness and security in cloud-based systems. Chapter 21 explores the software architectures for self-protection, while Chap. 22 discusses the benchmarks for intrusion detection systems. The final three chapters of Part IV extend the applications and use cases beyond the domain of data centers. Chapter 23 discusses the self-aware networks. Chapter 24 discusses the cyber-physical systems, with a special emphasis on run-time architectural models that may well apply broadly, beyond this specific application domain. Both Chaps. 21 and 24 can serve as practical complements to the more general and theoretical treatment of architecture in Part II. In the final chapter of Part IV, Chap. 25, we expand our horizons quite literally by discussing the vital role that self-aware computing plays in the control of autonomous spacecraft.

The final section of this book, Part V, consists of a single chapter on open challenges and future research directions for self-aware computing systems.

Intended Readership

This book is intended to serve two audiences. First, it may be used as a handbook for professionals and researchers who work in the areas related to self-aware computing. Second, it may be used as an advanced textbook for lecturers and students of postgraduate courses on any of the many subjects that relate to self-aware computing, such as advanced software engineering, autonomic computing, self-adaptive systems, data center resource management, and any of the other fields mentioned in Chap. 2. The reader is assumed to be generally acquainted with the principles and practices of computer science or software and systems engineering, but no specific expertise in any subfield of computer science is required. For the most part, each chapter is self-contained, and plenty of references are provided for those who wish to pursue the topic more deeply.

Würzburg, Germany
Hawthorne, NY, USA
Würzburg, Germany
Santa Clara, CA, USA
January 2017

Samuel Kounev
Jeffrey O. Kephart
Aleksandar Milenkoski
Xiaoyun Zhu

Contents

Part I Introduction

- 1 The Notion of Self-aware Computing. 3**
Samuel Kounev, Peter Lewis, Kirstie L. Bellman, Nelly Bencomo,
Javier Cámara, Ada Diaconescu, Lukas Esterle, Kurt Geihs,
Holger Giese, Sebastian Götz, Paola Inverardi, Jeffrey O. Kephart
and Andrea Zisman
- 2 Self-aware Computing Systems: Related Concepts
and Research Areas 17**
Javier Cámara, Kirstie L. Bellman, Jeffrey O. Kephart, Marco Autili,
Nelly Bencomo, Ada Diaconescu, Holger Giese, Sebastian Götz,
Paola Inverardi, Samuel Kounev and Massimo Tivoli
- 3 Towards a Framework for the Levels and Aspects
of Self-aware Computing Systems 51**
Peter Lewis, Kirstie L. Bellman, Christopher Landauer, Lukas Esterle,
Kyrre Glette, Ada Diaconescu and Holger Giese
- 4 Reference Scenarios for Self-aware Computing 87**
Jeffrey O. Kephart, Martina Maggio, Ada Diaconescu, Holger Giese,
Henry Hoffmann, Samuel Kounev, Anne Koziolk, Peter Lewis,
Anders Robertsson and Simon Spinner

Part II System Architectures

- 5 Architectural Concepts for Self-aware Computing Systems 109**
Holger Giese, Thomas Vogel, Ada Diaconescu, Sebastian Götz
and Samuel Kounev
- 6 Generic Architectures for Individual Self-aware
Computing Systems 149**
Holger Giese, Thomas Vogel, Ada Diaconescu, Sebastian Götz
and Kirstie L. Bellman

7	Architectures for Collective Self-aware Computing Systems	191
	Ada Diaconescu, Kirstie L. Bellman, Lukas Esterle, Holger Giese, Sebastian Götz, Peter Lewis and Andrea Zisman	
8	State of the Art in Architectures for Self-aware Computing Systems.	237
	Holger Giese, Thomas Vogel, Ada Diaconescu, Sebastian Götz, Nelly Bencomo, Kurt Geihs, Samuel Kounev and Kirstie L. Bellman	
Part III Methods and Algorithms		
9	Self-modeling and Self-awareness.	279
	Kirstie L. Bellman, Christopher Landauer, Phyllis Nelson, Nelly Bencomo, Sebastian Götz, Peter Lewis and Lukas Esterle	
10	Transition Strategies for Increasing Self-awareness in Existing Types of Computing Systems	305
	Marco Autili, Kirstie L. Bellman, Ada Diaconescu, Lukas Esterle, Massimo Tivoli and Andrea Zisman	
11	Synthesis and Verification of Self-aware Computing Systems	337
	Radu Calinescu, Marco Autili, Javier Cámara, Antiniscia Di Marco, Simos Gerasimou, Paola Inverardi, Alexander Perucci, Nils Jansen, Joost-Pieter Katoen, Marta Kwiatkowska, Ole J. Mengshoel, Romina Spalazese and Massimo Tivoli	
12	Self-adaptation for Individual Self-aware Computing Systems	375
	Martina Maggio, Tarek Abdelzaher, Lukas Esterle, Holger Giese, Jeffrey O. Kephart, Ole J. Mengshoel, Alessandro V. Papadopoulos, Anders Robertsson and Katinka Wolter	
13	Self-adaptation in Collective Self-aware Computing Systems.	401
	Jeffrey O. Kephart, Ada Diaconescu, Holger Giese, Anders Robertsson, Tarek Abdelzaher, Peter Lewis, Antonio Filieri, Lukas Esterle and Sylvain Frey	
14	Metrics and Benchmarks for Self-aware Computing Systems	437
	Nikolas Herbst, Steffen Becker, Samuel Kounev, Heiko Kozirolek, Martina Maggio, Aleksandar Milenkoski and Evgenia Smirni	
15	Assessing Self-awareness.	465
	Lukas Esterle, Kirstie L. Bellman, Steffen Becker, Anne Kozirolek, Christopher Landauer and Peter Lewis	

Part IV Applications and Case Studies

16 Run-Time Models for Online Performance and Resource Management in Data Centers	485
Simon Spinner, Antonio Filieri, Samuel Kounev, Martina Maggio and Anders Robertsson	
17 Online Learning of Run-Time Models for Performance and Resource Management in Data Centers	507
Jürgen Walter, Antinisca Di Marco, Simon Spinner, Paola Inverardi and Samuel Kounev	
18 Online Workload Forecasting	529
Nikolas Herbst, Ayman Amin, Artur Andrzejak, Lars Grunske, Samuel Kounev, Ole J. Mengshoel and Priya Sundararajan	
19 State of Practice of Non-self-aware Virtual Machine Management in Cloud Data Centers	555
Lydia Y. Chen, Robert Birke and Evgenia Smirni	
20 Self-awareness of Cloud Applications	575
Alex Iosup, Xiaoyun Zhu, Arif Merchant, Eva Kalyvianaki, Martina Maggio, Simon Spinner, Tarek Abdelzaher, Ole Mengshoel and Sara Bouchenak	
21 Software Architectures for Self-protection in IaaS Clouds	611
K.R. Jayaram, Aleksandar Milenkoski and Samuel Kounev	
22 Benchmarking Intrusion Detection Systems with Adaptive Provisioning of Virtualized Resources	633
Aleksandar Milenkoski, K.R. Jayaram and Samuel Kounev	
23 Self-aware Networks: The Cognitive Packet Network and Its Performance	659
Erol Gelenbe	
24 Leveraging Design and Runtime Architecture Models to Support Self-awareness	669
Philippe Lalanda, Stéphanie Chollet and Catherine Hamon	
25 Spacecraft Autonomous Reaction Capabilities, Control Approaches, and Self-aware Computing	687
Klaus Schilling, Jürgen Walter and Samuel Kounev	

Part V Outlook

**26 Self-aware Computing Systems: Open Challenges
and Future Research Directions. 709**
Robert Birke, Javier Cámara, Lydia Y. Chen, Lukas Esterle,
Kurt Geihs, Erol Gelenbe, Holger Giese, Anders Robertsson
and Xiaoyun Zhu

Contributors

Tarek Abdelzaher Computer Science Department, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Ayman Amin Swinburne University of Technology, Melbourne, Australia

Artur Andrzejak Heidelberg University, Heidelberg, Germany

Marco Autili University of L'Aquila, L'Aquila, Italy

Steffen Becker Technical University Chemnitz, Chemnitz, Germany; Aerospace Corporation, El Segundo, CA, USA

Kirstie L. Bellman Aerospace Integration Science Center, The Aerospace Corporation, Los Angeles, CA, USA; Topcy House Consulting, Thousand Oaks, CA, USA; Technische Universität Chemnitz, Chemnitz, Germany

Nelly Bencomo Aston Institute for Systems Analytics, Aston University, Birmingham, UK

Robert Birke IBM Research Zurich, Zürich, Switzerland

Sara Bouchenak INSA Lyon, Lyon, France

Radu Calinescu University of York, York, UK

Javier Camara Carnegie Mellon University, Pittsburgh, PA, USA

Lydia Y. Chen IBM Research Zurich, Zürich, Switzerland

Stéphanie Chollet Grenoble University LCIS, Valence, France

Javier Cámara Carnegie Mellon University, Pittsburgh, PA, USA

Ada Diaconescu Equipe S3, Département INFRES, Télécom ParisTech, Paris, France; Telecom ParisTech, CNRS LTCI, Paris Saclay University, Paris, France

Lukas Esterle Department of CS, Vienna University of Technology, Vienna, Austria; Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria; Technische Universität Wien, Vienna, Austria

Antonio Filieri Department of Computing, Imperial College London, London, UK

Sylvain Frey Computing and Communications Department, Lancaster University, Lancaster, UK

Kurt Geihs University of Kassel, Kassel, Germany

Erol Gelenbe Department of Electrical and Electronic Engineering, Imperial College, London, UK

Simos Gerasimou University of York, York, UK

Holger Giese Hasso Plattner Institute for Software Systems Engineering at the University of Potsdam, Potsdam, Germany

Kyrre Glette University of Oslo, Oslo, Norway

Lars Grunske Humboldt-Universität zu Berlin, Berlin, Germany

Sebastian Götz University of Technology Dresden, Dresden, Germany

Catherine Hamon Orange Labs, Meylan, France

Nikolas Herbst University of Würzburg, Würzburg, Germany

Henry Hoffmann Department of Computer Science, University of Chicago, Chicago, IL, USA

Paola Inverardi University of L'Aquila, L'Aquila, Italy; University of L'Aquila, Coppito (AQ), Italy

Alex Iosup Delft University of Technology, Delft, The Netherlands

Nils Jansen University of Texas at Austin, Austin, TX, USA

K.R. Jayaram IBM Thomas J. Watson Research Center, New York City, NY, USA

Eva Kalyvianaki Imperial College of London, London, UK

Joost-Pieter Katoen RWTH Aachen University, Aachen, Germany

Jeffrey O. Kephart IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA

Samuel Kounev Department of Computer Science, University of Würzburg, Würzburg, Germany

Anne Koziolk Institute for Program Structures and Data Organization, Karlsruhe Institute of Technology, Karlsruhe, Germany

Heiko Koziolk ABB Ladenburg, Ladenburg, Germany

Marta Kwiatkowska University of Oxford, Oxford, UK

Philippe Lalanda Grenoble University LIG, Grenoble, France

Christopher Landauer Aerospace Integration Science Center, The Aerospace Corporation, Los Angeles, CA, USA; Topcy House Consulting, Thousand Oaks, CA, USA; Aerospace Corporation, El Segundo, CA, USA

Peter Lewis Aston Lab for Intelligent Collectives Engineering (ALICE), School of Engineering and Applied Science, Aston University, Birmingham, UK

Martina Maggio Department of Automatic Control, Lund University, Lund, Sweden

Antiniscia Di Marco University of L'Aquila, L'Aquila, Italy

Ole Mengshoel CMU Silicon Valley at the NASA Ames Research Center, Moffett Field, PA, USA

Ole J. Mengshoel Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, CA, USA

Arif Merchant Google, Inc., Menlo Park, CA, USA

Aleksandar Milenkoski University of Würzburg, Würzburg, Germany

Phyllis Nelson California State Polytechnic University, Pomona, CA, USA

Alessandro V. Papadopoulos Department of Automatic Control, Lund University, Lund, Sweden

Alexander Perucci University of L'Aquila, L'Aquila, Italy

Anders Robertsson Department of Automatic Control LTH, Lund University, Lund, Sweden

Klaus Schilling Department of Computer Science, University of Würzburg, Würzburg, Germany

Evgenia Smirni College of William and Mary, Williamsburg, VA, USA

Romina Spalazzese Malmö University, Malmö, Sweden

Simon Spinner Department of Computer Science, University of Würzburg, Würzburg, Germany

Priya Sundararajan Carnegie Mellon University, Pittsburgh, CA, USA

Massimo Tivoli University of L'Aquila, L'Aquila, Italy

Thomas Vogel Hasso Plattner Institute for Software Systems Engineering at the University of Potsdam, Potsdam, Germany

Jürgen Walter Department of Computer Science, University of Würzburg, Würzburg, Germany

Katinka Wolter Institute of Computer Science, Freie Universität Berlin, Berlin, Germany

Xiaoyun Zhu Futurewei Technologies, Santa Clara, CA, USA

Andrea Zisman The Open University, Milton Keynes, UK

**Matrix and Tensor Factorization Techniques for
Recommender Systems**

Symeonidis, P.; Zioupos, A.

2016, VI, 102 p. 51 illus., 22 illus. in color., Softcover

ISBN: 978-3-319-41356-3