The Knowledge Engineering Review, Vol. 30:4, 373–374. © Cambridge University Press, 2015
doi:10.1017/S0269888915000041

Introduction to the special issue on challenges in agent-oriented software engineering

DANNY WEYNS[1,2] and JÖRG P. MÜLLER[3]

[1]*Department of Computer Science, Katholieke Universiteit Leuven, Belgium;*
e-mail: danny.weyns@cs.kuleuven.be;
[2]*Department of Computer Science, Linnaeus University, Sweden;*
e-mail: danny.weyns@lnu.se;
[3]*Department of Informatics, Technische Universität Clausthal, Germany;*
e-mail: joerg.mueller@tu-clausthal.de

1 Introduction

The upcoming generation of software systems will increasingly consist of loosely coupled interacting subsystems. Examples are management systems for the power grid, service systems for business collaborations, large-scale traffic control systems, and networked smart homes. Engineering these systems and guaranteeing the required qualities (performance, robustness, etc.) during system operation is complex due to the inherently decentralized nature of these systems as well as uncertainties resulting from incomplete knowledge at design time. Among the uncertainties are subsystems that come and go at will, dynamically changing availability of resources, and faults that are difficult to predict. The body of knowledge developed in the multi-agent systems engineering research community under the label of agent-oriented software engineering will be invaluable to tackle the engineering challenges of the next-generation software systems.

The goal of this special issue is to provide an overview of the state of the art in different key areas of engineering multi-agent systems, with particular attention being devoted to challenges for future research and the position of the work in the broader field of agent-oriented software engineering and software engineering in general. Authors of selected papers from the Agent-Oriented Software Engineering Workshop 2011 were invited to submit revised and extended versions of their workshop publications. The submitted papers went through a rigorous multiple-staged review process; finally four high-quality papers were selected for publication.

2 Overview of the issue

The papers of the special issue cover four key topics in the space of engineering multi-agent systems: agent-oriented software engineering methodologies, programming of multi-agent systems, software engineering for self-organizing systems, and coordination infrastructures for decentralized systems.

2.1 Agent-oriented software engineering methodologies

Gomez-Sanz and Fuentes-Fernandez start their paper from the observation that the focus of research efforts in the agent-oriented software engineering community have been too narrow, not recognizing the kind of development methods that industry demands. To that end, the authors propose a definition of what an agent-oriented software engineering methodology should constitute. Based on this definition, the authors perform a review of a selection of methodologies, identifying missing elements and opportunities for future lines of improvement. The authors identify different challenges to push the field forward, including agent technology and methodologies should be applied to industry-size development projects;

https://doi.org/10.1017/S0269888915000041 Published online by Cambridge University Press

missing activities need to be integrated to enhance the scope of the methodologies; and empirical approaches for agent-oriented software engineering are needed.

2.2 *Programming of multi-agent systems*

In his paper, Dastani provides an overview of the area of multi-agent programming focusing on both programming languages and development frameworks. In the first part of the paper, the author discusses the key abstractions that need to be addressed when programming multi-agent systems—agents, organizations, and environments—and explores different ways of implementing them. The second part connects abstractions with realization approaches. The author summarizes the state-of-the-art languages and supporting frameworks for programming multi-agent systems. From this analysis two main trends are identified: support for different and more expressive goal types and support for modular programming. Dastani concludes with discussing two key challenges: the integration of programming languages for individual agents, organizations, and environments; and support for debugging and testing of multi-agent programs.

2.3 *Software engineering for self-organizing systems*

Parunak and Brueckner start with pinpointing the distinction between three kinds of systems: autonomous, self-adaptive, and self-organizing. Subsequently, the authors survey a selection of work in the field of self-organizing systems and their applications. From this review, four general trends are identified: decentralization, openness, imitation of nature, and reliance on simulation. From an analysis of these trends, the authors highlight a number of research challenges, including the problem of composing more complex systems, the challenge of characterizing and controlling an existing system, and finally the objective of understanding self-organizing systems formally. The authors conclude with a discussion how the engineering of self-organizing systems has drawn much from the engineering of conventional software.

2.4 *Coordination infrastructures for decentralized systems*

Rodrıguez-Aguilar *et al*. review state-of-the-art coordination infrastructures aiming to identify open research challenges that next-generation coordination infrastructures should address. The analysis identifies the following key challenges for coordination infrastructures: (i) to become socially aware by facilitating human interaction within the multi-agent system; (ii) to assist agents in their decision making by providing decision support that helps them reduce the scope of reasoning and facilitates the achievement of their goals; and (iii) to increase openness to support online, fully decentralized design and execution. The authors outline promising approaches for the different challenges.

This summary demonstrates that tackling the challenges of engineering complex decentralized software systems is a multi-faceted endeavor. There remains plenty of room for researchers to contribute to this undertaking. We offer the papers of this special issue as a benchmark on the current state of the art, and an exposition of key ideas and directions for further work.

Acknowledgments

We thank all the authors for submitting their work to the special issue. We are grateful to the reviewers for their excellent work. Finally, we thank *KER* for hosting the special issue and, in particular, the editors-in-chief for their support throughout the process of preparing it.

https://doi.org/10.1017/S0269888915000041 Published online by Cambridge University Press