Softw Syst Model (2013) 12:435–438
DOI 10.1007/s10270-013-0339-3

EDITORIAL

Building European software architecture community: how far have we come?

Muhammad Ali Babar · Ian Gorton · Flavio Oquendo

Published online: 17 April 2013

1 Introduction

Whilst the references to the role and importance of software architecture started appearing since 1969 when NATO organized a conference on software engineering techniques, the term "software architecture" started gaining acceptance by the software engineering community in early 1990s. In 1992, Perry and Wolf [9] provided an earlier framework for grounding software architecture as an important area of research and practice. A few years later in 1996, Shaw and Garlan [11] published one of the first books on software architecture. Less than a year later, Bass et al. [2] published a practice-oriented book on software architecture. Whilst these efforts play a significant role in promoting software architecture as discipline of research and practice, the foundations of this discipline included seminal work of Edsger Dijkstra, David Parnas, and others between 1960s and 1980s [8]. There was also an increasing realization that an architectural description can play an important role in successfully understanding and managing large and complex software systems [3]. The need and importance of providing appropriate support for getting the architecture right were becoming quite visible. It had been felt that software architecture plays a fundamental role in setting the basis for the required quality attributes such as maintainability, reliability, usability, performance, and flexibility of a system [2].

Consequently, the last decade of the twentieth century witnessed an enormous amount of research activities aimed at strengthening and maturing the nascent discipline of software architecture. As a result of continuously and concentrated efforts by researchers and practitioners, software architecture has become a well-established area of research and practice by the middle of first decade of twenty-first century [5,10,12]. By then a young but active community of software architecture researchers and practitioners had emerged through community building platforms such as Working IEEE/IFIP Conference on Software Architecture (WICSA) and IFIP WG 2.10 Software Architecture. Kruchten et al. [5] provided a succinct summary of the achievements of the software architecture community by 2006. Whilst European researchers and practitioners had made significant contributions to the efforts to promote the software architecture research, practice, and education during the time period that Shaw and Clements called "The Golden Age of Software Architecture" [10], there was an important need of having a European platform for building a dedicated community to support and promote software architecture as an important sub-discipline of software engineering.

European Workshop on Software Architecture (EWSA) was founded in 2004 [7] by Oquendo, Warboys, and Morrison, the Principal Investigators of the ArchWare European project on Software Architecture, partially supported by the EU's Research Framework Program. The key purpose of this workshop was to provide researchers and practitioners with a platform to present and discuss the most recent, innovative

M. A. Babar
Lancaster University, Lancaster, UK

M. A. Babar (✉)
IT University of Copenhagen, Copenhagen, Denmark
e-mail: maba@itu.dk
URL: http://malibabar.wordpress.com/

I. Gorton
Pacific Northwest National Laboratory, Richland, USA
e-mail: ian.gorton@pnl.gov
URL: http://www.linkedin.com/in/gortonator/

F. Oquendo
IRISA, University of South Brittany, Vannes, France
e-mail: flavio.oquendo@irisa.fr
URL:http://www-irisa.univ-ubs.fr/flavio.oquendo/

and significant findings and experiences in the field of software architecture research and practice. There were two more editions of successful EWSA in 2005 (Pisa, Italy) and 2006 (Nantes, France) before the European community decided to convert the workshop into a full-fledged conference from 2007 and named the conference, European Conference on Software Architecture (ECSA), which was firstly organized in 2007 (Madrid, Spain) [6]. Succeeding increasingly successful editions were held in 2008 (Paphos, Cyprus), 2009 (Cambridge, UK; jointly organized with the 8th Working IEEE/IFIP Conference on Software Architecture, WICSA), 2010 (Copenhagen, Denmark), 2011 (Essen, Germany), and 2012 jointly with the 10th WICSA in Helsinki, Finland. The 2013 edition will be held in Montpellier, France and the 2014 in Vienna, Austria.

From the viewpoint of building a European community of software architecture research, practice, and education, several milestones have been achieved. ECSA has become a premier software architecture conference, providing a stimulating environment for presenting and discussing innovative ideas, state-of-the-art solutions, and futuristic research agendas in the area of software architecture research, practice, and education. The work presented through ECSA platform has focused on addressing the most important challenges in the area of software architecture such as architectures for adaptive and dynamically configurable systems, design rationale and architectural knowledge management, techniques for formal and semi-formal description and verification of architecture, service-oriented architecture, agile and lean approaches to architecting, and architectural solutions for model-driven development. European researchers are playing leading roles for several ongoing activities and initiatives to promote and support software architecture research, practice, and education.

Furthermore, ECSA has built over the years a strong European community for addressing the new challenges paving the way for horizon 2020 [4]. Definitely, in this second decade of twenty-first century, software architecture has been increasingly playing a key role for enabling the next generation of software-intensive systems. Software architecture forms the backbone for taming the complexity of software-intensive systems, in particular when these complex systems operate in open-world environments enabled by pervasive Internet and cloud computing, such as in smart cities, smart farms, and smart grids.

Beyond systems, the grand challenge is to address emergent systems-of-systems. Hence, the endeavor of constructing software-intensive systems evolved from architecting 'complicated' systems in the last decades, to architecting 'complex' systems-of-systems in this new one. Their upcoming generation will operate in environments that are open in the sense that they are only partially known at design time. These open-world systems-of-systems, in opposite to current closed-world systems, will run on dynamically discovered devices and networks.

Some of the current and emerging challenges in the area of software architecture have been raised and addressed in the articles included in this special issue of the *Journal of Software and Systems Modeling* (SoSyM). The four articles included in this special issues have been selected for significant extension and revision from the papers published in the 4th edition of the European Conference on Software Architecture [1] and its workshops. In the 2010 edition, apart from receiving more than 100 submissions under three different categories for the main conference, the collocated workshops also received and accepted a large number of high-quality papers. For the main conference, the organizing committee accepted only 19 full papers of 75 full papers submitted. The acceptance rate for the full papers was 25 % for ECSA 2010. In 2010, the conference featured three keynotes, research track, industrial track, workshops, tutorials, doctoral symposium, and tool demonstrations and poster presentations. Though the conference's name suggests a focus on Europe, ECSA has become one of the mainstream conferences of software architecture community and attracts a large proportion of its contributions and attendees from all over the World. For example, the 2010 edition of ECSA attracted papers (co-) authored by researchers, practitioners, and academia from 30 different countries (Algeria, Australia, Austria, Belgium, Brazil, Canada, Chile, China, Colombia, Czech Republic, Denmark, Finland, France, Germany, Hong Kong, Iceland, India, Ireland, Israel, Italy, the Netherlands, Poland, Portugal, Romania, Spain, Sweden, Switzerland, Tunisia, UK, and the USA).

2 The articles in this issue

The four papers that appear in this special issue attempt to answer to some of the most important challenges of software architecture research and practice faced today and over the coming years.

The paper "Modeling customer-centric value of system architecture investments" by Ana Ivanovic, Pierre America, and Chris Snijders focuses on the challenges involved in taking a stakeholder-centric approach to design and evolve software architectures of large-scale complex systems. Their work has been motivated by an important need of improving the alignment between software architecture design decisions and current and future business goals of an organization. The article reports an approach to model customers' value to support the decisions for choosing among proposed architectures of software-intensive systems. The presented approach is based on authors' experiences from two case studies of software architecture of healthcare systems. The stakeholders' value has been modeled by leveraging the theoretical concepts and practical techniques from management and marketing such as strategy maps, balanced scorecards, customer value-in-use and customer segments. Overall this article presents a set of useful tools to quantify the value of architecture changes for a single or multiple customers with several potential benefits.

Clones (i.e., duplicates) can cause several types of problems and one of them is elegant and smooth evolution. Most of the time clones are referred to code clones. However, there is an increasing realization that cloning can happen (and does happen) in other artifacts of a software system. Model clones can be quite troublesome to a model's quality, however, there has been no serious effort aimed at raising the awareness about model clones and addressing the challenges that can be associated with model clones. The paper "Towards clone detection in UML domain models" by Harald Störrle reports one of the very first attempts at targeting an important topic of detecting clones of models building using Unified Modeling Language (UML), which is also a de facto standard in software development industry. The author has thoroughly analyzed the practical scenarios to provide a definition of the notion of "model clone" and has presented a clone detection algorithm for UML domain models along with a prototype tool. The article presents and discusses different similarity heuristics for using the algorithm vis-à-vis the performance of the approach. The article also describes the potential limitations of the presented approach and the future plans to carry out more research and development to address the known limitations.

Dynamism is one of the key characteristics of the state-of-the-art software-intensive systems which are expected to be composed of autonomous and dynamically linked services following the principles of service orientation in software systems. One of the key challenges of designing and analyzing service-oriented dynamic systems is to model dynamic service-oriented architectures. The article "A model for dynamic reconfiguration in service-oriented Architectures" by José Luiz Fiadeiro and Antónia Lopes provides useful answers to some of the main challenges of modeling the dynamic characteristics of software architectures of systems. The presented work takes an approach that is aimed at going beyond what had been addressed by architecture description languages (ADLs). The work focuses on leveraging the logical separation between the service need and the need-fulfillment mechanism, i.e., the provision of the service. The authors have shown how the possibility of deferring the binding between the requester and the provider to run time and establishing it at the instance level, i.e., each time the need for the service arises, can be exploited to model dynamic reconfiguration that is general enough to support the definition of ADLs that are able to address the full dynamics of service-oriented applications. The authors have presented a simple service-oriented ADL derived from the modeling language that is being developed by authors and their colleagues.

The last paper in this special issue focuses on the security challenges that are needed to be considered and addressed during software architecture design. The article "Least privilege analysis in software architectures" by Koen Buyens, Riccardo Scandariato, and Wouter Joosen reports an interesting piece of work, which has been motivated by an increasing realization that some types of software security related issues should be precisely defined and effectively addressed during architecture design of a system. The authors assert that software architects tend to neglect security design principles. Such practice has the potential of high-risk threats to any software-intensive system. The article reports a piece of work that focuses on providing a formal foundation for understanding the security design principle of least privilege in software architectures. The authors also provide a technique to identify and/or analyze the violations against the presented security design principle. The reported approach to taking security design principles into consideration during software architecture design has been supported by appropriate tools, which have been validated in four case studies.

Acknowledgments We are grateful to the members of the software architecture community and the program committee of ECSA 2010 for helping us to seek a large number of submissions and provide valuable and timely reviews for the conference and for the papers included in this special issue. Their efforts enabled us to include high-quality papers in the ECSA 2010 proceedings and in this special issue. We are also thankful to all of those who submitted papers and/or attended ECSA 2010. We would also like to express our special gratitude to the reviewers for the papers which were invited for extension and revision for this special issue. We also would like to thank the SoSyM editorial office and, in particular, Martin Schindler, for the excellent support in preparing this theme issue.

References

1. Ali Babar, M., Gorton, I. (eds.): Proceedings of the 4th European conference on software architecture (ECSA 2010), Copenhagen, Denmark, August 23–26, 2010. LNCS, vol. 6285. Springer, Berlin
2. Bass, L., Clements, P., Kazman, R.: Software architecture in practice. Addison-Wesley, Menlo Park (1997)
3. Clements, P., Kazman, R., Klein, M.: Evaluating software architectures: methods and case studies. Addison-Wesley, Menlo Park (2002)
4. ISTAG, Towards horizon 2020, March 2012 http://cordis.europa.eu/pagination/break/fp7/ict/istag/documents/istag-wg-2013-v12-march-2012-new.pdf
5. Kruchten, P., Obbink, H., Stafford, J.: The past, present, and future of software architecture. IEEE Softw. **23**(2), 22–30 (2006)
6. Oquendo, F. (ed.): Proceedings of the 1st European conference on software architecture (ECSA 2007), Madrid, Spain, September 24–26, 2007. LNCS, vol. 4758. Springer, Berlin
7. Oquendo, F., Warboys, B., Morrison, R. (eds.): Proceedings of the 1st European workshop on software architecture (EWSA 2004), St Andrews, UK, May 21–22, 2004. LNCS, vol. 3047. Springer, Berlin

8. Parnas, D.L.: On the criteria to be used in decomposing systems into modules. Commun. ACM **15**(12), 1053–1058 (1972)
9. Perry, D.E., Wolf, A.L.: Foundations for the study of software architecture. ACM SIGSOFT, Software Engineering Notes **17**(4), 40–52 (1992)
10. Shaw, M., Clements, P.: The golden age of software architecture. IEEE Softw. **23**(2), 31–39 (2006)
11. Shaw, M., Garlan, D.: Software architecture: perspectives on an emerging discipline. Pearson Prentice Hall, Boston (1996)
12. Taylor, R.N., Medvidovic, N., Dashofy, E.M.: Software architecture: foundations, theory, and practice. Wiley, London (2009)

Author Biographies

Muhammad Ali Babar is a Reader in Software Engineering at Lancaster University, UK and an Associate Professor at IT University of Copenhagen, Denmark. Previously, he worked as a researchers and project leaders different research centers in Ireland and Australia. He has authored/co-authored more than 140 peer-reviewed research papers in journals, conferences, and workshops. He has co-edited a book, software architecture knowledge management: theory and practice. Dr. Ali Babar has been a guest editors of several special issues/sections of IEEE Software, JSS, ESEJ, and REJ. Apart from being on the program committees of several international conferences such as WICSA/ECSA, ESEM, SPLC, ICGSE, and ICSP for several years, Dr. Ali Babar has also chaired the program committees of several conferences such as WICSA/ECSA 2012, ECSA2010, PROFES2010, and ICGSE2011. He is a member of steering committees of WICSA, ECSA and ICGSE. He has presented tutorials in the areas of software architecture and empirical approaches at various international conferences including ICGSE09, XP08, ICSE07 and WICSA07. Prior to joining R&D field, he worked as a software engineer and an IT consultant for several years in Australia. His current research interests include cloud computing, software architecture, and global software development. He obtained a PhD in computer science and engineering from University of New South Wales, Australia.

Dr. Ian Gorton was a Laboratory Fellow in the Computational Sciences and Mathematics Division at the Pacific Northwest National Laboratory (PNNL). He leads projects in various scientific domains that are building tools to process and manage complex, massive, distributed data collections for modeling and simulation. He has recently moved to the Software Engineering Institute at Carnegie Mellon University.

Flavio Oquendo is Full Professor of Computer Science and Software Engineering (holding a Research Excellence Award from the French Ministry of Research and Higher Education) at the University of South Brittany, France. He is the Head of the UMR CNRS IRISA's Research Team on Software Architecture. He received the B.Eng. degree from ITA, Sao Paulo, and the M.Sc., Ph.D., and H.D.R. (Research Direction Habilitation) degrees from the University of Grenoble. Before 2005, he held positions as Full Professor at the University of Savoie and Associate Professor at the University of Grenoble. Prior to entering the academic career, he was R&D Staff Member of the G.I.E. Emeraude at the Bull Research Center in Paris. He has published over 150 refereed journal and conference papers in Computer Science and Software-intensive Systems Architecture and Engineering. He has served on Program Committees of over 100 International Conferences, e.g. ICSE, ESEC/FSE, has chaired 10 of them, of which the French, European, and IEEE/IFIP International Conferences on Software Architecture. He is also the ECSA Steering Committee Chair. He has been editor or co-editor of more than 15 journal special issues and books. In the European Research Area, he has actively participated in a dozen European Research Projects in several EU Framework Programs, of which over 10 years as Scientific Director. He has also acted as Expert, Reviewer and Evaluator for the European Commission. His research interests are centered on formal languages, processes and tools to support the efficient architecture of complex software-intensive systems and systems-of-systems. For more details: http://www-irisa.univ-ubs.fr/flavio.oquendo.