GUEST EDITORIAL

Introduction to the theme section on Agile model-driven engineering

Kevin Lano[1] · Shekoufeh Kolahdouz-Rahimi[2] · Javier Troya[3] · Hessa Alfraihi[4]

1 Introduction

Agile methods have become a widely used software development approach across many industry sectors [14], with established benefits in terms of increased responsiveness to change and decreased time to market. Agile practices emphasise lightweight and iterative development, designed to deliver value to customers quickly.

The model-driven approach to software development (MDE) originated at about the same time as Agile, in the late 1990s, and has been most widely utilised in high-integrity domains such as vehicle control systems and aerospace. It has the benefits of a rigorous and systematic approach to software construction, based on models which represent the key concepts of the application.

Because Agile and MDE both have attractive features, the idea arose of producing an integration of the two approaches, in order to gain the advantages of both. From the perspective of MDE researchers, incorporating Agile practices into MDE appeared to be a way to widen the uptake of MDE to more general software application areas. From the perspective of industrialists using Agile approaches, the introduction of MDE techniques was considered as a means of increasing the rigour and precision of their practices.

✉ Kevin Lano
  kevin.lano@kcl.ac.uk

  Shekoufeh Kolahdouz-Rahimi
  sh.rahimi@eng.ui.ac.ir

  Javier Troya
  jtroya@uma.es

  Hessa Alfraihi
  haalfraihi@pnu.edu.sa

1 Informatics Department, King's College London, London, UK

2 Department of Software Engineering, University of Isfahan, Isfahan, Iran

3 ITIS Software, Universidad de Málaga, Málaga, Spain

4 Department of Information Systems, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia

The concept of combining Agile and MDE has therefore been considered by researchers and practitioners from 2002 onwards, resulting in new integrated methods, and tools and industrial applications. Two research teams at European universities (King's College London and the University of Bremen) specialise in the topic, and major companies such as Motorola and Volvo have adopted Agile MDE approaches.

1.1 The challenges of combining Agile and MDE

While some aims of Agile and MDE are compatible, such as the aim to accelerate the development process, the approaches also have significant incompatibilities: MDE tends to use substantial documentation (models) produced in a staged manner, whilst Agile is focussed on the programming level and tries to use a minimum of documentation. The concept of multiple inter-related models or views of a system is a basic principle of the UML, and this can hinder agility because a change to one model (such as a state machine) may affect its consistency with respect to another model (such as a class diagram or interaction), leading to the need to change these. Graphical models tend to require more effort to create and edit than textual models. They are useful as an aid to understanding a system and its domain; however, for communication with customers, prototypes may be more generally suitable. The focus of MDE on tools whilst Agile focusses on close team and customer collaboration, is another point of potential incompatibility.

1.2 Early Agile MDE approaches 2000–2009

One of the first attempts to combine Agile and MDE was Scott Ambler's 'Agile modeling' approach in 2002 [2]. This combined practices from the XP agile method with the Rational Unified Process for UML-based MDE. Another approach was Stephen Mellor's 'Agile MDA' method [10], which focussed on using MDE models as executable artefacts, to which agile techniques could be applied in the same manner

as to conventional programs. This line of work has continued with the definition of an executable variant, fUML, of UML [13].

1.3 Advances and industrial application 2010–2020

More detailed and specific Agile MDE approaches began to emerge from industrial experience with major projects, in telecoms, business systems, web applications and automotive systems. The MDD-SLAP method of [15] integrates a Scrum-based agile approach with a V-model MDE process for the construction of telecoms systems. The method of [6] introduced 'meta-sprints' to organise activities and artefacts—such as business-level models—which endure beyond the scope of individual sprints. In the automotive sector, the industry-standard AUTOSAR method combines aspects of MDE and Agile development, and has been utilised by several major companies [4]. Agile MDE approaches have also been utilised in aerospace systems [11] and financial systems [8,12].

An important tendency that can be observed in the application-based work was the utilisation of relatively small and lightweight domain-specific modelling languages (DSLs) instead of complex UML modelling notations. This choice has an impact in reducing the complexity and costs of tooling for modelling support and hence in increasing agility.

Surveys of Agile MDE also began to appear, for example [1,5] together with books proposing specific Agile MDE methods [7]. Tools supporting agile modelling also appeared, such as Simulink [9] and AgileUML [3]. A workshop series XM ('eXtreme Modelling') was hosted at MODELS from 2012 to 2014 and was succeeded by the 'FlexMDE' workshop between 2015 and 2019. While the latter was not specific to Agile MDE, it was one of the main focusses of the workshops.

1.4 Current status

Full success in overcoming the challenges of integrating Agile and MDE cannot be claimed, but there have been a number of success stories in specific domains, with methods and modelling approaches tailored to particular application areas such as vehicle control systems and web applications. Indeed advances in the field of Agile MDE seem often to have come from insights arising from industrial practice, and our theme section reflects this situation, with two of the three featured papers focussed on the needs of industrial practitioners.

2 Selected papers for the theme section

Three papers were selected from the submissions to the theme section. In "Agile MERODE: a model-driven software engineering method for user-centric and value-based development", Yves Wautelet and Monique Snoeck define a combination of the MERODE MDE method with the Scrum agile method, producing an integrated method, Agile MERODE. The aim is to provide an MDE method driven by user stories. A case study shows the practical application of the method.

In "Using DevOps toolchains in Agile model-driven engineering", Joern Suess, Samantha Swift and Eban Escott identify the need for MDE to support Development and Operations (DevOps) practices in order to become more agile, including the use of continuous integration and deployment (CI/CD). They identify problems with current MDE tooling support for DevOps and give an industrial experience report describing how they enabled CI/CD pipelines using DevOps and MDE.

In "Consistency management in industrial continuous model-based development settings: a reality check", Robert Jongeling, Federico Ciccozzi, Jan Carlson and Antonio Cicchetti investigate the state of practice of consistency management in industrial MDE contexts. They find that practitioners seek to adopt shorter development cycles, but that there is deficient MDE support for such an approach. They identify the need for more automated consistency management, and ideally, support for continuous model-based development.

References

1. Alfraihi, H., Lano, K.: The integration of agile development and MDE: a systematic literature review. Modelsward (2017)
2. Ambler, S.: Agile Modeling: Effective Practices for Extreme Programming and the Unified Process. Wiley, Hoboken (2002)
3. Eclipse AgileUML project. https://projects.eclipse.org/projects/modeling.agileuml (2022)
4. Eliasson, U., Heldal, R., Lantz, J., Berger, C.: Agile MDE in mechatronic systems—an industrial case study. In MODELS 2014, LNCS, vol. 8767. Springer (2014)
5. Hansson, S., Zhao, Y., Burden, H.: How MAD Are We? Empirical Evidence for Model-Driven Agile Development. XM (2014)
6. Kulkarni, V., Barat, S., Ramteerthkar, U.: Early Experiences with Agile Methodology in a Model-Driven Approach. MODELS (2011)
7. Lano, K.: Agile Model-Driven Development with UML-RSDS. Taylor and Francis, Abingdon (2016)
8. Lano, K., Haughton, H., et al.: Agile Model-Driven Engineering of Financial Applications. FlexMDE, MODELS (2017)
9. Mathworks, Simulink toolset. https://uk.mathworks.com/products/simulink.html. (2022)

10. Mellor, S., Agile MDA. https://www.omg.org/mda/mda_files/Agile_MDA.pdf. (2005)

11. Mirachi, S., da Costa Guerra, V., Marques da Cunha, A., Vieira Dias, L., Villani, E.: Applying agile methods to aircraft embedded software. SPE **47**, 1465–1484 (2017)

12. Nakicenovic, M.B.: An agile driven architecture modernization to a model-driven development solution. Int. J. Adv. Softw. **5**(3), 308–322 (2012)

13. OMG. Semantics of a Foundational Subset for Executable UML Models (FUML), v1.1 (2015)

14. VersionOne. State of Agile Survey, www.explore.versionone.com/state-of-agile (2020)

15. Zhang, Y., Patel, S.: Agile model-driven development in practice. IEEE Software (2011)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.