Special issue on Model-driven Engineering of Multi-agent Systems in Theory and Practice

Geylani Kardas, Jorge J. Gomez-Sanz

DOI: 10.1016/j.cl.2017.07.002

To appear in: *Computer Languages, Systems & Structures*

Published online: 5 July 2017

Special issue on Model-driven Engineering of Multi-agent Systems in Theory and Practice

Geylani Kardas[1,*] and Jorge J. Gomez-Sanz[2]

[1] Ege University, International Computer Institute, 35100, Bornova, Izmir, Turkey
geylani.kardas@ege.edu.tr

[2] Universidad Complutense de Madrid, 28040, Madrid, Spain
jjgomez@ucm.es

Creating agents that autonomously interact to solve problems can be a challenging task. To cope with it, researchers in Multi-agent systems (MASs) have recently made extensive use of the Model Driven Engineering (MDE) paradigm.

Model-driven engineering (MDE) aims to change the focus of software development from code to models while enabling the systematic use of models as primary engineering artifacts throughout the engineering lifecycle. It starts with the definition of meta-models, which collect and interconnect relevant abstractions identified by the researchers. Then, they are used to define general purpose modeling languages (GPMLs) and domain-specific (modeling) languages (DSLs/DSMLs). Supporting computer-aided software engineering (CASE) tools are built upon them. Finally, development methodologies are created, identifying development processes where deliverables are produced with the assistance of CASE tools. Deliverables can be the result of the combination of different models, but also documentation, and programming code, perhaps produced automatically from the models.

MDE applied to autonomous agents and MASs has become an active research topic in the agent-oriented software engineering (AOSE) community. MDE has helped agent researchers to investigate which abstractions are needed to define a MAS and determine how they can be used to build one along a development cycle. For some time, researchers were limited to use GPMLs, such as UML. Advances in MDE has allowed researchers to go beyond, extending current GPMLs, and trying out new alternatives, creating brand new agent-oriented modeling

[*] Corresponding author. Tel: 90-232-3113223

languages, for instance. Moreover, formal specification of related agent DSL/DSML semantics paves the way to the verification of the constructed MAS models and the implementation of the systems on various agent execution platforms, if needed.

Besides, as new development activities are incorporated into the MAS development processes, it becomes necessary to formally define the scope of those activities, what they do, what they require, and what they produce. MDE can help in this endeavour too through the use of DSMLs oriented towards software process specification. This practice had led to a greater understanding and reuse of the development processes in MASs. In particular, research has produced meta-models of fragments for agent-oriented methodologies to, among others, propose new combinations of such fragments adapted to specific development needs.

This special issue of Computer Languages, Systems & Structures (COMLAN) journal aims at presenting recent studies on MDE to MAS along the abovementioned research directions. This includes MAS modeling, model-based MAS specification, model-driven MAS architectures, DSL/DSML specification, CASE tool development, and empirical evaluation of model-driven MAS development. We received 8 submissions and 3 of them were finally accepted for inclusion in the special issue after a rigorous peer-review process involving at least 3 reviewers per paper.

The papers included in this special issue can be summarized as follows:

- *Model Driven Approach for Real-time Requirement Analysis of Multi-Agent Systems* by Amir Ashamalla, Ghassan Beydoun, and Graham Low. The authors take the view that the earlier you model real-time requirements in the software development life cycle, the more reliable and robust the resultant system will be. Hence, a model-driven approach to MAS development is proposed that ensures real-time requirement constraints are taken into account during the analysis of a MAS. A meta-model is introduced to identify the real time requirements of a MAS before MAS design. The validation of the proposed meta-model is performed in a Call Management MAS application.
- *Agent-oriented model-driven development for JADE with the JADEL programming language* by Federico Bergenti, Eleonora Iotti, Stefania Monica, and Agostino Poggi. An agent-oriented programming language called JADEL is introduced in this paper.

JADEL is a DSL which facilitates model-driven MAS development on the well-known JADE platform. JADEL supports four core high-level abstractions, namely agents, behaviours, communication ontologies, and interaction protocols. JADEL is also supported with a compiler which enables source code generation for implementing JADE agents.

- *User-Story Driven Development of Multi-Agent Systems: A Process Fragment for Agile Methods* by Yves Wautelet, Samedi Heng, Soreangsey Kiv, and Manuel Kolp. The authors present a process fragment for the integration of agent-based development in agile methods based on User Stories sets. User Stories are operational requirements representation models and they can be used in agile methods to drive a software transformation approach embedded in a particular development paradigm. An initial set of User Stories are used in the study to build a reasoning model, called the Rationale Tree. A Rationale Tree can then be aligned with a multi-agent design and implemented in an agent-oriented development language. The process is modeled as a process fragment that contributes to the requirements analysis, MAS design and MAS implementation phases.

As editors of this special issue, we think the selected papers will contribute to advancing the state-of-the-art in model-driven engineering applied to Multi-Agent Systems. They can be of assistance to those readers investigating the relevant future research directions on MDE in MAS. Also to practitioners that look for pragmatic approaches to solve problems where autonomously interacting entities are expected to appear. All of them can find convincing examples of MDE applied to MAS development in the selected papers.

We would like to thank all reviewers who participated in the reviewing process of the papers submitted to this special issue. Finally, we would also like to thank COMLAN journal's Editor-in-Chief, Marjan Mernik, for his strong support and help he provided in all stages of preparing this special issue.