A Formal Approach for the Modeling of High-Level Architectures Aligned with System Requirements

Racem Bougacha^(✉)

Institut de Recherche Technologique Railenium, 59300 Famars, France
racem.bougacha@railenium.eu

1 Problem Statement and Motivations

IRT Railenium (<http://railenium.eu/fr/>) is a test and applied research center for the rail industry in France. One of its three R&D and innovation programs aims in particular to provide the technological tools and bricks necessary for the development of the Autonomous Train. This Autonomous Train program will thus address signaling, control-command, driving and railway operating systems. The Autonomous Freight Train project under the Autonomous Train program with the cooperation of several partners (“SNCF”, “ALSTOM”, “Hitachi Rail STS France”, “ALTRAN” and “APSYS”) targets performance improvements of the system thanks to the implementation of autonomy in railways operations. This system is classified as Cyber-Physical System (CPS) and depends more and more on effective solutions that can address heterogeneity and interplay of physical and software elements. In particular, modeling languages used for specifying CPS should incorporate, in a consistent manner, the essential concepts from multiple engineering disciplines that take part in the design of such systems.

In response to this need, our work addresses (1) the lack of a common modeling language between these disciplines, which can hamper reasoning about system properties. This issue is stemming from, on the one hand, several modeling languages specified for designing CPS that have been standardized but none of them provide the full range required to deal effectively with the heterogeneity of CPS elements. On the other hand, complex systems such as autonomous freight trains may include many concerns from different modeling formalisms.

A second issue addressed in our work is (2) the non-existence of a holistic approach for designing Autonomous Freight Trains from a high-level architecture to a formal specification where it can be possible to check the compliance with system requirements. Therefore, this issue could be divided into two sub-issues. The first sub-issue is deduced from the fact that architecture models actually used are semi-formal and/or informal, so their specifications are still no valid because they are not proved. Thus, formal specifications are needed in order to guarantee the consistency and the completeness of architecture models specification.

The quality of a system is the main measure of its success, that depends on the degree to which it fulfills its requirements. Requirements modelling is an important activity in the process of designing and managing high-level architectures of systems. From this context the second sub-issue is raised, i.e alignment links between requirements models, domain models and architecture models should be established. These semantic links can be the support to prove the compliance of an architecture specification with the expression of system requirements.

From this perspective, we formulate the following research questions:

- **RQ1:** Can we provide a modeling language for High-Level architectures of Autonomous Freight Train?
- **RQ2:** What are the criteria for defining such a language?
- **RQ3:** How can we provide a formal specification of High-Level Architectures to verify its consistency?
- **RQ4:** How can we establish and verify alignment links between High-Level Architectures and System Requirements?

2 Related Work

Several architecture modeling languages have been proposed to reason about heterogeneous properties of CPS, and assessing multidisciplinary design languages. The authors of paper [1] propose an approach to combine SysML [2] and AADL [3]. These formalisms are two modeling languages specified for designing Integrated Control Systems (ICSs). These languages have been standardized but none of them provides the full range required to deal effectively with more general kinds of complex systems like CPS. In fact, SysML supports requirements engineering, traceability, and modeling of diverse physical phenomena. On the other hand, AADL is oriented to model real-time embedded systems. It provides software-to-hardware bindings allowing analyses of different system properties such as performance, timing, etc. This combination consists in extending SysML using the UML extension mechanism **profiles** to cover all AADL concepts and is called **Extended SysML for Architecture Analysis Modeling (ExSAM)**. This approach allows to design high-level architectures of complex systems. However, in this approach only semi-formal graphical models can be produced and thus no formal verification can be carried out. Furthermore, they do not permit to specify alignment links that can be used to ensure the compliance of architecture models with system requirements.

Authors in [5] proposes a multi approach for real-time systems specification and design. The purpose of the work is to couple MARTE [4] with the Event-B method. Papers [6] and [7] consider a meaningful subset of AADL which allows to specify respectively a class of embedded control systems and AADL data port protocols, and assign this subset a semantics in terms of Event-B refinements and model decomposition. Despite these approaches provide formal specifications, they do not consider high-level architectures but propose a temporal model of tasks execution on GPU.

Requirements Engineering (RE) is defined as the branch of software engineering which is focused on real-world goals, requirements development and management. [8] presents a review on Goal-Oriented Requirements Engineering (GORE) methods. This kind of methods defines goals as objectives that the system under consideration should achieve. [9] and [10] present an approach which aims to combine requirements engineering methods with formal methods. The main idea is to specify goal models by using a SysML-based language [2] extended with concepts of the KAOS goal language [8], to specify domain models by using an ontology-based language, and then to map them into an Event-B specification. However, these methods are just concerned by requirements models and do not provide formalism to specify system architecture models.

3 The Proposed Approach and Methodology

We propose to define a method for modeling high-level architectures for autonomous freight trains that integrates concepts from the different engineering disciplines that take part in the design process of such systems. The method needs to be multiviews: graphical in order to be validated by all the stakeholders and formal in order to verify both architecture models and their compliance with system requirements. This is a three-steps method based on the MDE approach shown in Fig. 1. In the first step, we propose to combine three graphical modeling languages: SysML, AADL and MARTE, in order to provide the full range required to deal effectively with Autonomous Freight Train concerns such as timing properties, non-functional properties, software-to-hardware bindings, etc. The second step consists in formalizing the concepts of the graphical language with Event-B in order to obtain an Event-B specification of architecture models. Thus their correctness and consistency can be proved. We have chosen the Event-B method since it is a formal method, widely used in industry, based

Fig. 1. The proposed approach

on refinement and decomposition mechanisms and supported by tools (provers, model-checkers, animators...). In the final step, we propose to specify alignment links between requirements models and high-level architecture models in order to prove that all the industrial stakeholders requirements are satisfied by high-level architecture models. We will reuse the SysML/KAOS method to produce requirements models (goal and domain models) and the corresponding Event-B specifications.

4 Current Assessment and Future Work

The current focus of this doctoral work, which is in its first year, is on the design of a graphical language for modeling high-level architectures. This first step attempts to answer RQ1 and RQ2 (step 1 of Fig. 1). The proposed modeling approach will be applied on industrial case studies. For instance we will consider the standard logical system architectures composed of three subsystems: on-board subsystem, trackside subsystem and communication subsystem, from the Autonomous Freight Train project. These case studies will be validated by the project partners (system, sub-system and requirements engineers) and verified by design models tools (for instance AADL verification tools) and Event-B verification tools (for instance ProB model checker and Atelier B prover).

References

1. Behjati, R., Yue, T., Nejati, S., Briand, L., Selic, B.: An AADL-based SysML profile for architecture level systems engineering: approach, metamodels, and experiments. ModelME! report, 2001-03 (2011)
2. OMG systems modeling language (2005). <http://www.omgsysml.org/>
3. Feiler, P.H., Gluch, D.P., Hudak, J.J.: The architecture analysis & design language (AADL): an introduction (No. CMU/SEI-2006-TN-011). Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA (2006)
4. OMG, UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded systems, Beta 2 (2008)
5. Zouaneb, I., Belarbi, M., Chouarfia, A.: Multi approach for real-time systems specification: case study of GPU parallel systems. IJBDE 3(2), 122–141 (2016)
6. D'Souza, M., Ramesh, S., Satpathy, M.: Architectural semantics of AADL using Event-B. In 2014 International Conference on Contemporary Computing and Informatics (IC3I), pp. 92–97. IEEE (2014)
7. Filali-Amine, M., Lawall, J.: Development of a synchronous subset of AADL. In: Frappier, M., Glässer, U., Khurshid, S., Laleau, R., Reeves, S. (eds.) ABZ 2010. LNCS, vol. 5977, pp. 245–258. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-11811-1_19
8. Van Lamsweerde, A.: Goal-oriented requirements engineering: a guided tour. In: Proceedings Fifth IEEE International Symposium on Requirements Engineering, pp. 249–262. IEEE (2001)
9. Matoussi, A., Gervais, F., Laleau, R.: A goal-based approach to guide the design of an abstract Event-B specification. In: ICECCS 2011, pp. 139–148 (2011)

10. Tuono Fotso, S.J., Mammar, A., Laleau, R., Frappier, M.: Event-B expression and verification of translation rules between SysML/KAOS domain models and B system specifications. In: Butler, M., Raschke, A., Hoang, T.S., Reichl, K. (eds.) ABZ 2018. LNCS, vol. 10817, pp. 55–70. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91271-4_5