



HAL
open science

MegaM@Rt2 EU Project: Latest Advances in Megamodelling at Design Time and Runtime for Cyber-Physical Systems

Hugo Bruneliere, Dragos Truscan, Pierluigi Pierini, Andrey Sadovykh

► **To cite this version:**

Hugo Bruneliere, Dragos Truscan, Pierluigi Pierini, Andrey Sadovykh. MegaM@Rt2 EU Project: Latest Advances in Megamodelling at Design Time and Runtime for Cyber-Physical Systems. 2020. hal-03121170

HAL Id: hal-03121170

<https://hal.science/hal-03121170v1>

Preprint submitted on 26 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - ShareAlike 4.0 International License

MegaM@Rt² EU Project: Latest Advances in Megamodelling at Design Time and Runtime for Cyber-Physical Systems

Hugo Bruneliere
IMT Atlantique, LS2N (CNRS)
Nantes, France

hugo.bruneliere@imt-atlantique.fr

Dragos Truscan
Åbo Akademi University,
Finland

dragos.truscan@abo.fi

Pierluigi Pierini
Intecs Solutions S.p.A.
Italy

pierluigi.pierini@intecs.it

Andrey Sadovkyh
Innopolis University
Russia

a.sadovkyh@innopolis.ru

ABSTRACT

MegaM@Rt² is a large ECSEL-JU European project whose main objective is to propose a scalable model-based framework incorporating methods and tools for the continuous development and runtime support of complex software-intensive Cyber-Physical Systems (CPSs). In this context, a fundamental challenge notably resides in providing efficient forward and backward traceability support between the two main system levels: i.e. design-time and runtime. To this extent, significant research and development efforts have already been realized in MegaM@Rt² in order to address relevant problems from three main complementary areas: System Engineering, Runtime Analysis, and Model & Traceability Management. In this paper, we summarize the MegaM@Rt² conceptual approach and give an overview of the latest research results obtained during these past months. Moreover, we discuss some important related scientific challenges as well as how we intend to address them.

CCS Concepts

- Software and its engineering → Software creation and management → Software verification and validation
- Computing methodologies → Modeling and simulation → Model development and analysis → Modeling methodologies
- Software and its engineering → Software creation and management → Software development process management → Software development methods.

Keywords

Cyber-Physical Systems; Model-Driven Engineering; Runtime Analysis; Tools; Mega-Modelling; Traceability; ECSEL

1. INTRODUCTION

MegaM@Rt² is a three-years project, funded by European Components and Systems for European Leadership Joint Undertaking (ECSEL JU) under the H2020 European program, which started in April 2017 [1], [2]. The main goal of MegaM@Rt² is to create an integrated framework incorporating scalable methods and tools for continuous system engineering and runtime validation and verification (V&V) which addresses the needs of the 8 case study providers involved in the project, which come from diverse and heterogeneous industrial domains, ranging from transportation and telecommunications to logistics and manufacturing.

In order to address these needs, 20 technology and research providers provide more than 28 tools to be integrated into the MegaM@Rt² framework with the underlying objective to provide improved productivity, quality, and predictability of large and complex industrial cyber-physical systems (CPSs). In this paper, we provide a brief overview of the latest results of the MegaM@Rt² project covering the third and last year of the project. The results enumerated in this paper are a complement to the ones discussed in [3].

The paper is structured as follows. Section 2 introduces the major objectives of MegaM@Rt² and describes the overall conceptual architecture behind it. The focus is set on the three main complementary areas targeted in the project, namely System Engineering, Runtime Analysis, and Model & Traceability Management. Section 3 presents in detail the latest research results obtained in the project. These research contributions are grouped together according to the MegaM@Rt² scientific area they mostly belong to. However, it is interesting to note that some of the mentioned works are actually cross-areas even though conveniently attached to a particular area in the paper. Finally, Section 4 concludes the paper.

2. MEGAM@RT² OVERALL CONCEPT

The MegaM@Rt² project focuses on offering a scalable model-based framework incorporating methods and tools for the continuous development and runtime support of complex software-intensive CPSs. A fundamental challenge notably resides in providing efficient forward and backward traceability support between the two main system levels: i.e. design-time and runtime. In parallel to these, modern large-scale industrial software engineering processes require thorough configuration and model governance to provide the promised productivity gains. As an answer to that, MegaM@Rt² intends to provide a scalable megamodelling approach to manage all the involved artifacts including the multitude of different types of models, corresponding workflows, and configurations, among others. In this context, an important challenge is to better tackle their large diversity in terms of nature, number, size, and complexity.

More globally, the MegaM@Rt² project brings together research labs and tool providers who come with different state-of-the-art methods and tools. The main outcomes of this project are being deployed and validated in real use cases from the European industry. This way, actual end users from the space, naval, railway, smart grid, smart warehouse, and telecom industries play

the central role of providing realistic requirements and corresponding case studies.

Fig. 1 provides an overview of the MegaM@Rt² global approach. It describes its main principles and related core concepts. In their daily operation, industries apply a set of heterogeneous engineering practices based on various modelling methodologies, languages/standards and tools: for example, SysML, AADL, EAST_ADL, Matlab/Simulink, AUTOSAR, Method B, and Modelica. Each one of these solutions results in the creation of numerous specific requirement specifications and design models, which are then used more or less automatically to produce the final software and hardware artefacts.

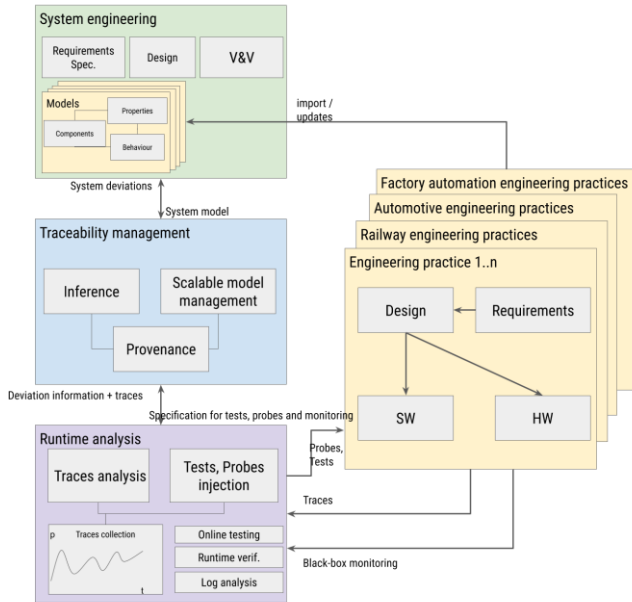


Figure 1. Conceptual architecture of the MegaM@Rt² project

In MegaM@Rt², we propose to integrate all those artefacts, and notably the modelling ones, in order to provide a complete view of a CPS. This view contains details of the components, behaviors, and desired quality properties of the engineered system. The specifications of these properties can then be used as inputs of a continuous testing and monitoring process of the runtime environment. The objective is to detect possible design deviations in real-time and to trace them back to the impacted design components. When possible, automatic corrections and improvements can also be suggested to rectify and enhance (and then redeploy) the system. Interestingly, this also triggers the next iteration of the overall continuous integration process.

In the following sections, we report on the latest contributions and results from the MegaM@Rt² project in a direct relation to the above-described conceptual picture.

3. LATEST RESULTS OF THE MEGAM@RT²

This section briefly summarizes the latest results of the MegaM@Rt² project while providing pointers to complementary scientific publications describing them in more detail. More generally, and in addition to the results obtained within the different technical work packages (cf. the following subsections), we also recently published the main outcomes of our joint effort on proposing a tool-supported model-based approach for building architectures and roadmaps [4]. Indeed, in the context of the

MegaM@Rt² project, we elaborated on a pragmatic model-driven approach to specify the case study requirements, design the high-level architecture of a framework, perform a gap analysis between the industrial needs and current state-of-the-art, and plan a first framework development roadmap accordingly. In this paper, we discuss the collaborative modelling process, the requirement definition tooling, the approach for components modelling, as well as the provided traceability and document generation. Although we have applied this solution only in the context of MegaM@Rt² project, what we propose in this work is completely generic. It can be adapted and applied in the context of many other projects. Finally, the paper ends with a discussion on the practical lessons we have learned from this work so far.

3.1 System Engineering Featuring Mega-Modelling at Runtime

In this first area, the community addresses the development of system modelling methods and tools for creating, verifying, and validating the design level models in the context of continuous development strategy addressed by MegaM@Rt² project. The main objectives of the research and development efforts provided by the MegaM@Rt² technology partners aimed to extend and improve several aspects of the modelling methodologies.

One of the main outcomes is the contribution to the MARTE 2.0 standard, answering to the OMG RFI [5]. The partners' contributions highlight the current lack of MARTE, among others, in: a) security formal semantics to describe security requirements and mechanisms, attacker's views and timing interferences; b) enhancement for performance, schedulability and timing analysis (e.g. multi-clocks behaviors, modelling demand of hardware (HW) and software (SW) resources, resource scheduling policy, workload generators); c) extended support for contract-based design and safety issues.

The contract-based approach is exploited on safety related systems proposed in railway and automotive domain use cases (UCs) applying the space domain customization presented in [6]. Similarly, to space systems, systems belonging to other domains often need to be engineered in compliance with standards, such as the European Cooperation for Space Standardization (ECSS), and need to ensure a certain degree of dependability. Given the multi-faceted nature of dependability (characterized by a set of concerns), assuring dependability implies multi-concern assurance, which requires modelling of various system characteristics and their co-assessment and co-analysis, in order to enable the management of trade-offs between them. Haider et al. [6] presented the customization of the CHESSE methodology and tool (included in MegaM@Rt² framework) and its Failure Logic Analysis (FLA) techniques to perform dependability-centered co-analysis for assuring the required non-functional properties of the system according to ECSS requirements.

The variability modelling is another important feature required by several use cases to manage product lines. Bilic et al. [7] described an Integrated Model-based Tool Chain for Managing Variability in Complex System Design based on the Modelio tool. Such variability approach allows to annotate SysML models with variability data exchanged between the system modelling tool and variability management tools through the Variability Exchange Language. The paper includes the introduction of the model-based product line engineering tool chain and its application on a practical UC at Volvo Construction Equipment. Initial results suggest an improved efficiency in developing such a variable system.

A combination of variant management and contract-based design for safety-critical systems is investigated in [8] and [9]. Contracts are envisaged to support compositional verification of a system as well as independent development and reuse of their implementations. But reuse of safety-relevant components in safety-critical systems need to cover more than just the implementations. As many safety-relevant artefacts related to a component should also be reused together with the implementation to assure that the system under development is acceptably safe.

For verification and validation of models at design time, Iqbal et al. [10] investigated exhaustive model simulation and model-based test generation using fUML activity diagrams. The paper shows how design specifications can be used to generate tests using executable UML models. The approach generates test cases along with test data and test oracle in order to left shift the testing process. The test data are also used to simulate the design models to improve the specifications by means of early detection and rectification of design faults. Additionally, the work presented in [10] also includes code generation and reuse of test cases. The presented approach generates a code skeleton for the design models which can be later reused as a boilerplate for system development along with the generated test suite to validate the implementation, and to verify the system requirements. The proposed approach is implemented as an Eclipse plugin called MATERA2-ALF Tester [11].

Considering the model-based design methodologies and frameworks developed or enhanced as part of Meg@MaRT2, the work in [12] focuses on an electronic-system level design space exploration environment for dedicated heterogeneous multi-processor embedded systems that exploits a customized SystemC simulator to identify feasible implementation alternatives. The work presents the main design aspects related to the SystemC library extension introduced to support the Alternation concept [13]. Moreover, it describes the main SW design issues related to the developed design space exploration tool, and the full integration of the main tools in the context of the whole SystemC-based co-design flow. Finally, some reference UCs are proposed in order to verify and validate the whole methodology.

Starting from the co-design methodology presented in [12], the work in [14] presents a HW/SW co-design framework for mixed-criticality embedded systems able to consider, in the final implementation, also Xtratum-based SW partitions [15] optimized to fulfil safety and partitioning constraints. The framework has been implemented into an Eclipse-plugin (called HEPSYCODE), and use a SystemC simulator, Java Architecture for XML Binding (JAXB) technology, and XML schema with marshal transformation to perform project and code generation.

Finally, Dragomir et al. [16] presented the refinement calculus of reactive systems toolset, which provides an environment for compositional formal modelling and reasoning about reactive systems, built around Isabelle, Simulink, and Python. The toolset implements the Refinement Calculus of Reactive Systems (RCRS), a contract-based refinement framework inspired by the classic Refinement Calculus and interface theories. The toolset formalizes the entire RCRS theory in about 30000 lines of Isabelle code. The toolset also contains a translator of Simulink diagrams, and a formal analyzer implemented on top of Isabelle. The main functionalities of the RCRS toolset are presented via a series of pedagogical examples. The work in [16] also presents a larger case study from the automotive domain.

3.2 Runtime Analysis

In the runtime analysis area, MegaM@Rt² deals with code generation, runtime monitoring, analysis of execution logs, data clustering, and functional and non-functional testing. The latest MegaM@Rt² results presented in this section include a performance testing approach, a systematic study on model-based testing, and a novel clustering method.

Ahmad et al. [17] presented an exploratory performance testing approach called PerfXRL, which uses reinforcement learning to find performance bottlenecks in a black-box system without any prior domain knowledge. In this approach, a learning agent explores a large space of the input value combinations and focuses only on those areas of the input space which are most likely to trigger performance bottlenecks in the system under test. The results presented in this paper show that, after executing only 25% of the total combinations of input values, PerfXRL managed to find approximately 72% more performance bottlenecks than a random testing technique. To automate the exploration process, Ahmad et al. [17] provide adequate tool support for their approach.

Software-based systems are permeating the ever-increasing number of business areas starting from web applications to systems used to manage critical functions of different devices and safety-critical systems, for instance, pacemakers, aircraft, and nuclear power plants. Therefore, such systems must meet the quality criteria required by the application domain and expected by their customers. In order to identify the state-of-the-art software testing approaches that use UML Activity Diagrams (ADs), Ahmad et al. [18] performed a systematic mapping study. The objective of this study was to present a comprehensive and unbiased overview of the existing model-based testing approaches that use UML ADs. They posed nine research questions, outlined the selection criteria, and developed a classification scheme. They highlighted the current trends and research gaps in model-based testing using UML ADs and discussed some shortcomings for researchers and practitioners working in this area. The results of this study show that the existing approaches on model-based testing using UML ADs tend to rely on intermediate formats and formalisms for model verification and test generation, employ a multitude of graph-based coverage criteria, and use graph search algorithms.

In clustering, one attempts to identify the similarities of the data points and place them into collections according to the identified similarities. Numerous of the existent clustering algorithms establish a clustering structure regardless of whether the data entails such a structure. Therefore, the algorithms impose a structure to the data. An alternate approach is to attempt to discover a structure from the data, whether the data contains a structure at all. Jokinen et al. [19] proposed a novel clusterability assessment method called density-based clusterability measure, which evaluates the clustering structure in the data to assess whether a cluster analysis could actually present knowledge of the relationships in the data. Within the MegaM@Rt² project, this novel method has been successfully applied on the Nokia UC, in which the data comprises time-series data. The quantification of the structure provided by this clusterability measure presents a significant addition to the process of cluster analysis. This quantitative clusterability approach initially automatically identifies whether the data is clusterable or not. If the data is clusterable, the approach can automatically present knowledge on the correct number of clusters which are statistically identified. The incorporation of quantitative clusterability approach to the

overall process of clustering is proven beneficial. Finally, this approach can automatically and correctly provide knowledge on the clustering structure of the data in question.

3.3 Model Management and Traceability

In this area the project deals with the design and development of generic methods and tools for supporting global model and traceability management tasks in the context of complex CPS engineering processes. More particularly, the focus is set on creating and effectively supporting feedback loops between runtime and design time (cf. the work presented in the two previous subsections) in order to improve such overall processes. In what follows, we describe our latest research achievements related to such topics in the context of the MegaM@Rt² project.

As introduced earlier, complex systems are now predominant in several domains such as automotive, health-care, aerospace, industrial control, and automation. Such systems call for modern engineering practices such as Modelling/MDE to tackle advances in productivity and quality of these now CPSs. However, the proposed solutions need to be further developed to scale up for real-life industrial projects and to provide significant benefits at execution time. To this extent, one of the major challenges we intend to tackle in MegaM@Rt² is to achieve a more efficient integration between the design and runtime aspects of the concerned systems: The system behavior at runtime has to be better matched with the original system design in order to be able to understand critical situations that may occur, as well as corresponding potential failures in design. Methods and tools already exist (many of them not model-based) for monitoring system execution and performing measurements of runtime properties. However, they do not usually allow a relevant integration with (and/or traceability back to) design models. Such a feedback loop from runtime is highly relevant at design time, which is the most suitable level for system engineers to analyze and take impactful decisions accordingly.

To stimulate research efforts intending to overcome these challenges, we recently organized and successfully run (in July 2019) a second edition of the MDE@DeRun workshop [20], co-located with the STAF 2019 conferences. For the second time, following the first edition of MDE@DeRun in 2018 [21], this event provided a venue where both researchers and practitioners (i.e. engineers) on model-driven and/or model-based techniques and architectures for complex systems can meet. The purpose of this event is to better disseminate and exchange ideas or challenges, identify current/future key issues and explore possible solutions related to the above-mentioned family of problems.

This year's edition of the MDE@DeRun workshop notably included a special keynote by Professor Manuel Wimmer (Johannes Kepler University Linz, Austria) entitled "From Design-Time to Runtime and Back Again with Liquid Models". This keynote demonstrated once again the importance and relevance of runtime ↔ design time traceability and interactions in the context of current and future industrial systems. The event also included an invited talk by Dr Hugo Bruneliere (IMT Atlantique & LS2N-CNRS, France) entitled "A Model-based Framework for Continuous Development and Runtime Validation of Complex Systems: MegaM@Rt² EU Project Results". In this talk, we notably emphasized on the various already obtained MegaM@Rt² research results directly related to our work on the above-mentioned topics [3].

As part of this second edition of the MDE@DeRun workshop, we also presented a novel reference case study for design-runtime interaction in MDE [22]. This case study is based on Train Ticket,

a micro-service-based web application, and its associated monitoring infrastructure. It provides a reusable set of software modelling artifacts designed in UML and reverse engineered from the application source code, a dataset of related monitoring logs, and the definition of design-runtime example correspondences as traceability links. With this work, we invite researchers from the community to consider this case study as a reference for proposing future contributions to this topic. Thanks to this initiative, we hope to stimulate further research in this area.

As a final result of this second edition of the MDE@DeRun workshop, several open research questions and challenges were raised during the interactive discussions that occurred with the audience. They can be summarized as follows:

- More **abstractions** are needed to deal with the different kinds of "trace" data (e.g. execution logs, inter-model traceability data);
- The **scalability** of model-based approaches must be improved when it comes to jointly manipulate both trace and runtime data;
- Other important dimensions of runtime data must also be better taken into account, including for example the **temporality, representation or reliability**;
- Possible **misalignments** between design and runtime data (models) are a critical problem to be better handled as well (e.g. model synchronization aspects);
- Complementary, **user profiles** actually in charge of all these different activities have to be explicitly identified and supported;
- The development of **Digital Twins**, and their potential relation with (or impact on) design time-runtime traceability, must also be further investigated.

Moreover, in addition to these research efforts and brainstorm on traceability management, we can also report on our latest work from the model management side. Indeed, as we have seen, the scalability of modelling solutions is still an important aspect required by use cases implying the storage of large models, such as automatically generated ones (e.g. application logs, reverse engineered source code). To this end, we notably explored the mapping of conceptual schemas expressed in UML/OCL into a set of logical schemas, either relational or NoSQL ones, to be used to store (sometimes very) large application data according to a given partition [23]. In the proposed approach, designers can select the set of databases that better matches their requirements (e.g. in terms of scalability) and automatically translates their high-level models into corresponding database constructs. The integration of the different databases is done in a generic way via the proposed approach. In addition, our mapping covers as well the database queries required to implement and check various kinds of model constraints.

4. CONCLUSIONS

In this paper, we presented an overview of the latest research achievements realized in the context of the MegaM@Rt² collaborative European project. The described contributions cover three main complementary areas which are fundamental for the model-based engineering of large and complex CPSs: system design and modelling, model and traceability management, and runtime analysis. While the MegaM@Rt² project itself is ending, there are still many open challenges within each of these different areas. Thus, we already plan to push further our research and innovation efforts in the context of future collaborations and projects with various partners from the MegaM@Rt² consortium.

5. ACKNOWLEDGMENTS

This work has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737494. This Joint Undertaking receives support from the EU's Horizon 2020 research and innovation programme and Sweden, France, Spain, Italy, Finland, the Czech Republic.

6. REFERENCES

- [1] W. Afzal, H. Bruneliere, D. Di Ruscio, A. Sadovykh, S. Mazzini, E. Cariou, D. Truscan, J. Cabot, A. Gomez, J. Gorrongoitia, L. Pomante, P. Smrz. "The MegaM@Rt2 ECSEL project: MegaModelling at Runtime – Scalable model-based framework for continuous development and runtime validation of complex systems". *Microprocessors and Microsystems: Embedded Hardware Design (MICPRO)*, Elsevier, vol. 61, pp. 86–95, 2018.
<https://dx.doi.org/10.1016/j.micpro.2018.05.010>
- [2] "MegaM@Rt2 Project Website." [Online]. Available: <https://megamart2-ecsel.eu/>. [Accessed: 01-November-2020]
- [3] A. Sadovykh, D. Truscan, W. Afzal, H. Bruneliere, A. Ashraf, A. Gomez, A. Espinosa, G. Widforss, P. Pierini, E. Fournieret, A. Bagnato. "MegaM@Rt2 Project: Mega-Modelling at Runtime - Intermediate Results and Research Challenges". *Software Technology: Methods and Tools (TOOLS 2019)*, Oct 2019, Innapolis, Russia. *Lecture Notes in Computer Science*, vol 11771, pp.393-405. Springer, Cham. https://doi.org/10.1007/978-3-030-29852-4_33
- [4] A. Sadovykh, W. Afzal, D. Truscan, P. Pierini, H. Bruneliere, A. Bagnato, A. Gomez, J. Cabot, O. Avila-Garcia. "On a Tool-Supported Model-Based Approach for Building Architectures and Roadmaps: The MegaM@Rt2 Project Experience". *Microprocessors and Microsystems: Embedded Hardware Design (MICPRO)*, Elsevier, vol. 71, pp.102848, 2019.
<https://dx.doi.org/10.1016/j.micpro.2019.102848>
- [5] MARTE 2.0 RFI,
https://www.omg.org/schedule/MARTE_2.0_RFI.html
- [6] Z. Haider, B. Gallina, A. Carlsson, S. Mazzini, S. Puri, "ConcertoFLA-based Multi-concern Assurance for Space Systems", *Ada User Journal*, 40:1, March 2019.
- [7] D. Bilic, E. Brosse, A. Sadovykh, D. Truscan, H. Bruneliere, U. Ryssel. "An Integrated Model-based Tool Chain for Managing Variability in Complex System Design". *Models and Evolution Workshop (ME 2019)*, co-located with the IEEE / ACM 22nd International Conference on Model Driven Engineering Languages and Systems (MODELS 2019), Sep 2019, Munich, Germany.
<https://dx.doi.org/10.1109/MODELS-C.2019.00045>
- [8] I.Sljivo, B.Gallina, J.Carlson, H.Hansson, S.Puri, "Tool-Supported Safety-Relevant Component Reuse: From Specification to Argumentation", 23rd International Conference on Reliable Software Technologies (Ada-Europe 2018)
- [9] M.A. Javed, B.Gallina, A.Carlsson, "Towards variant management and change impact analysis in safety-oriented process-product lines", 34th ACM/SIGAPP Symposium on Applied Computing (SAC'19), Apr 2019
- [10] J. Iqbal, A. Ashraf, D. Truscan, I. Porres, "Exhaustive Simulation and Test Generation Using fUML Activity Diagrams". In: P. Giorgini, B. Weber (Eds.), *Advanced Information Systems Engineering, Lecture Notes in Computer Science 11483*, 96–110, Springer, Cham, 2019.
- [11] J. Iqbal, A. Ashraf, D., I. Porres, "MATERS2-Alf Tester: An Exhaustive Simulation and Test Generation Tool for fUML Models", 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pages: 466-470, 2019.
- [12] L. Pomante, V. Muttillio, M. Santic, P. Serri, *SystemC-based Electronic System-Level Design Space Exploration Environment for Dedicated Heterogeneous Multi-Processor Systems*, Microprocessors and Microsystems, 2019
- [13] C. A. R. Hoare. 1985. *Communicating Sequential Processes*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA
- [14] V. Muttillio, L. Pomante, P. Balbastre, J. Simo, and A. Crespo, "HW/SW Co-Design Framework for Mixed-Criticality Embedded Systems considering Xtratum-based SW Partitions," *Euromicro Conference on Digital System Design (DSD)*. 2019
- [15] A. Crespo, I. Ripoll, M. Masmano, "Partitioned Embedded Architecture Based on Hypervisor: The XtratuM Approach," *European Dependable Computing Conference*, 2010, pp. 67–72.
- [16] I. Dragomir, V. Preoteasa, S. Tripakis. "The Refinement Calculus of Reactive Systems Toolset". In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2018)*, p. 201-208, Springer International Publishing, Cham, 2018.
- [17] T. Ahmad, A. Ashraf, D. Truscan, and I. Porres, "Exploratory performance testing using reinforcement learning", 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pages 156–163, Aug 2019.
- [18] T. Ahmad, J. Iqbal, A. Ashraf, D. Truscan, I. Porres, "Model-based testing using UML activity diagrams: A systematic mapping study", *Computer Science Review*, Volume 33, 2019, Pages 98-112, ISSN 1574-0137,
- [19] J. Jokinen, T. Rätty and T. Lintonen, "Clustering structure analysis in time-series data with density-based clusterability measure," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 6, pp. 1–12, Nov. 2019.
- [20] A. Bagnato, H. Bruneliere, L. Burgueño, R. Eramo, A. Gomez. "STAF 2019 Co-Located Events Joint Proceedings: 1st Junior Researcher Community Event, 2nd International Workshop on Model-Driven Engineering for Design- Runtime Interaction in Complex Systems, and 1st Research Project Showcase Workshop co-located with Software Technologies: Applications and Foundations (STAF 2019), Eindhoven, The Netherlands, July 15 - 19, 2019.
- [21] H. Bruneliere, R. Eramo, A. Gomez, V. Besnard, J.M. Bruel, M. Gogolla, A. Kastner, A. Rutle. "Model-Driven Engineering for Design- Runtime Interaction in Complex Systems: Scientific Challenges and Roadmap". *MDE@DeRun 2018 workshop*, co-located with the Software Technologies: Applications and Foundations (STAF 2018) federation of conferences, Jun 2018, Toulouse, France. https://dx.doi.org/10.1007/978-3-030-04771-9_40
- [22] D. Di Pompeo, M. Tucci, A. Celi, R. Eramo, "A Microservice Reference Case Study for Design- Runtime Interaction in MDE," *MDE@DeRun workshop*, STAF 2019, pp. 23-32.
- [23] G. Daniel, A. Gómez, J. Cabot, *UMLToNoSQL: Mapping Conceptual Schemas to Heterogeneous Datastores*. *IEEE 13th International Conference on Research Challenges in Information Science (RCIS)*, 2019.