

Towards MARTE++: an enhanced UML-based language to Model and Analyse Real-Time and Embedded Systems for the IoT age

Julio L. Medina⁺, and Eugenio Villar^{*}

⁺Software Engineering and Real-Time Group, ^{*}Microelectronics Engineering Group, University of Cantabria, Spain.
{julio.medina, eugenio.villarb}@unican.es

Abstract— This paper presents requirements for an enhanced version of the UML Profile for MARTE, the current standard of the OMG for the modelling and analysis of real-time embedded systems. Since its adoption by the OMG in 2009 and after the various additions along recent years, MARTE has been essayed in a number of application domains and validation approaches. This paper makes a review of these various efforts describing extensions, additional functionality, and modeling needs that may serve as inputs for the preparation of a formal request for proposals (RFP) at the OMG. Aspects that have been found useful to have in it include modern platforms like Multi-core, Many-core and GPUs, networking for broader domains like the Internet of Things, federation of all modelling artifacts involved in the development process, including tracing mechanisms embedded in the language to link design and run-time artifacts, and more elaborated kinds of quantitative analyses and extra functional properties, like energy and memory consumption, heat dissipation, and temperature distribution. Also methodological aspects like its specification as a profile and/or as a meta-model will need to be discussed. Finally, the standard needs to be reviewed against the new executable UML related specifications; particularly to be in alignment with those that fix the semantics of state machines and composite structures.

Keywords— UML; MARTE; model-driven development; mixed criticality systems; embedded systems; real-time systems; extra-functional properties; software engineering; OMG Standards.

I. INTRODUCTION

Since its initial adoption by the OMG in 2009, the UML Profile for MARTE [1] has been used in, and adapted to several application domains and analysis approaches [2]. After this time, new areas, tools, technologies, and related specifications have emerged around it, suggesting the need for its adaptation to the modern practitioners' environments.

The effort presented here aims to contribute to the discussion about requirements that are to be included in the preparation of an official Request for Proposals (RFP) [3] at the OMG for a new version of MARTE. This is not an exhaustive list of the potential needs of the community, but states those we have identified so far. This short position paper presents a selection of aspects in the broad field of design and development of real-time and embedded systems, which have been found necessary in the usage of the UML Profile for MARTE and may require its extension or modification. Besides the currently unresolved

issues [4] posted by the community to the OMG Revision Task Force for MARTE, other modeling standards have been highlighting aspects that deserve attention of the OMG standardization community. Particularly those that pursue an executable UML set of specifications need to be considered carefully. This alignment to executable UML is in our opinion essential to keep MARTE as a valuable asset in the model driven development of real-time and embedded systems.

Looking around the specific domain of embedded systems, there are a number of additional tendencies in the field that should be taken into account. One of those is the severe slowing down of Moore's law. The stabilization of semiconductor manufacturing technologies will make the amortization of investments in each new line spread on longer periods of time so that manufacturing costs can be significantly reduced [5]. This may shift investments into high-level services and stability will bring more mature hardware and software platforms, for which MDE will also better amortize efforts spent in model transformations. Global technical and standardization challenges like the construction of cyber-physical systems (CPS) and trustable systems of systems (CPSoS), as well as matching components for the Internet-of-Things (IoT) post also concerns on a common language for their specification, design, and validation.

II. REQUIREMENTS FOR A NEW MARTE

This section describes the concerns that motivate this effort. Our intention is to trigger the discussion in the community and look for partners that may share our aim for enhancements.

A. Modern Platforms and Mixed criticalities

The increasingly higher computation capacity and lower price of modern multi-core and many-core platforms has gained the attention of the usually more conservative designers of embedded systems. Even sectors that usually deploy applications over micro-controllers and certified processors are considering using general purpose modern processors as processing resources for embedded systems with larger applications running on top.

This is the reasoning for initiatives like AUTOSAR [6] and the increasing need for mixed-criticalities systems [7] in the avionics and aero-space domains. As pointed out by IBM in MARTE issue #27300 [8], MARTE should support the

specification of hardware platforms that include multiple cores. This extent must include not only the hardware description but also the modern scheduling algorithms for multicore, including the capacity of tasks to migrate from one core to another as well as the mechanisms to do the core assignment (affinity) to tasks.

Criticality is a designation of the level of assurance against failure needed for a system component. A mixed criticality system is one that has two or more distinct levels (consider for example safety critical, mission critical and non-critical). Reviewing the standards in the field (IEC 61508, DO-178B, DO-254 and ISO 26262 standards) they propose to use up to five levels. Then, in general an integer value represented with `NFP_Integer` is sufficient to annotate the criticality level for a value or a constraint [7] [9].

Regarding the annotation of criticalities of NFP constraints, mix-criticality systems have the need to include allocated (or assigned) in the same model, representing either hardware or software platforms, elements with different levels of criticality. Eventually, for different levels of criticality different versions of the modelling element may need to be used. To handle this multi-versions of modelling elements as well as to indicate the level of criticality at which a concrete condition expressed in a constraint must hold, there is the need to attach a level of criticality to the `NFP_Constraint` element in the MARTE profile.

Regarding the annotation of criticalities of NFP values, mix-criticality systems have the need to host for a concrete magnitude (e.g. WCET) different values, each corresponding to a different level of criticality.

B. High-level application modelling semantics for analyzability and xUML conformance.

One of the main risks for real-time systems designed with UML is the absolute freedom of the designer to use constructs and patterns that can easily lead to actual systems whose timing predictability is completely impossible to analyze. The HLAM clause in MARTE was meant to help in this concern. Unfortunately, the semantics of the way the behaviors in `RtUnits` and `PpUnits` are activated is not explicitly included in MARTE. Several methodologies are proposed [10] [11] to keep in synch the executable generated code with the corresponding analysis models. It is very important that whichever constrained design environment is defined for MARTE (take for example [12]), its semantics gets aligned with the new standards that complement the current trend towards an executable UML (fUML, PSCS, and PSSM) [13] [14]. This is critical to ensure that any prediction done with analysis models, that claim to be faithful to the high-level design models semantics, can be truly respected by the semantics implemented in the code properly generated from such design models. Other specific models of computation might also be defined as valid variants for MARTE, but at least the one in the current executable UML should be supported.

Correspondingly the roadmap for those executable UML related specifications will also need to be aligned with the one for the new version of MARTE. Additionally, both of them need to be in tune with their base foundation, UML. Besides, SysML is also a relevant standard in the field with which all mentioned specifications shall be in synch. The new initiative for a SysML 2.0 is a nice opportunity to harmonize these specifications.

C. General purpose and ad-hoc networking

One of the outstanding opportunities but also biggest challenges for the traditional embedded systems design techniques is to accommodate for their interconnection through general purpose and/or unreliable networks. Eventually, traditional resource account oriented distributed systems may scale well for this purpose, but only in very controlled industrial environments. The Internet of Things and the emergent behavior in cyber-physical systems of systems are design and validation environments on which MARTE needs a boost. Some initial efforts have been produced [15] to enhance its suitability to model combined situations, but many more distributed modelling use cases need to be envisioned, formalized and assessed.

In particular the exploitation of cloud-based services and platforms for reconfiguration, and pre-analyzed environment identification, or even predictability analysis can be categorized.

A conceptual formalization of operational stages for devices, motes, components, nodes, sub-systems, systems, and larger composites may well help practitioners to specify their models at the right level of abstraction and envisioned functionality.

There are two concrete potential new elements in MARTE that can serve to model the overheads and the reliability of the network. These are the concepts of `NetworkInterface` and `AbstractChannel`. They serve to model real-time as well as general purpose networks. These are useful from both, modeling and analysis perspectives, but they fit better into the Generic Quantitative Analysis Modeling sub-profile.

D. Additional analysis and validation domains

From its inception, the generic quantitative analysis modeling (GQAM) sub-profile in MARTE was meant to accommodate more and various kinds of behavioral based quantitative analyses. Some efforts have been produced regarding dependability [16], but many other related topics can require specific modelling needs. Profiling cache usage for worst-case execution time calculation [17] and bounding stack utilization for example are among those analyses partly supported in MARTE that might be formally extended with specific analysis sub-profiles.

Other extra-functional properties, like power consumption, temperature space distribution, heat dissipation, cost, inner complexity, criticality compatibility at allocation, security impact, etc. may have specific sub-profiles specializations of GQAM.

Finally, contract based design, and its assessment may benefit from an extension of GQAM to non-behavioral but static structural model validation, both for deployment (allocation) assessment and cloud-based remote operational discovery and running environment validation.

Considering a complete embedded system modeling and design methodology approach like the one proposed in [18], a holistic framework for iteratively exploring the design space of complex, multi-processing, HW/SW embedded systems can be found. The specification methodology proposed there is SW centric, but includes platform independent model (PIM), platform description model (PDM), and the architectural

mappings to be explored leading to the platform-specific models (PSM). It also requires the definition of the design space and design restrictions and the modeling of the different environment scenarios. Generation tools produce executable models required for the performance analysis of the experiments selected by the design-space exploration tool. This goal required extending the flexibility provided by MARTE in the definition of the PSM. That effort indicate that various more complex kinds of allocation/assignment need to be explored in MARTE.

E. Hardware and Software synthesis

Considering the effects of the end of Moore's Law, semiconductor technology could become much more accessible to companies with smaller volumes. In the near future, the design of integrated circuits will be accessible to any company in which the value added by silicon compensates the higher non-recurrent engineering costs. This implies that the synthesis of high level models or the combined transformations of functional models with the intermediate models of computation, could be implemented on both, hardware and software. This additional freedom implies an extended design space, which may need additional constraints and extra-functional properties to be fully explored. Even the concepts of allocation and assignment managed in MARTE might need to be revisited. Links to hardware description languages, and again richer kinds of allocation will have to be explored.

In [19], MARTE has been used to support the modeling, analysis and synthesis of embedded systems. There the main technical challenges was the efficient design of complex, multi-processing, systems addressing the parallelization of the application code, the SW synthesis and the run-time power management. A MARTE specification methodology supporting parallelization and the automatic generation of the SW stack in each SMP node of the heterogeneous, multi-processing platform, is proposed. By changing the properties in the communication channels in the MARTE model it is possible to decide the degree of parallelism and segmentation. That effort brought also a tool (eSSYN), able to generate automatically the code to be deployed on each computational resource from the Platform-Specific Model (PSM). The lack of a platform-independent programming language at that time required associating different functional codes to each component mapped to different computational devices (i.e. C/C++ for CPUs and OpenCL for GPUs). The tool implemented is able to support several communication and synchronization middleware APIs such as MCAP, as well as the generation of multiple executables on different computing resources.

F. Managing views and tracing mechanisms along the development process and iterative validation strategies.

It is clear that most of the possible solutions to manage views, traces along the development process, and tagging of models for their use in any particular way are severely influenced by the methodological guidelines followed by the modeler. Hence, it is discouraged to try to get them as part of the language. Nevertheless, any generic strategy that may help to link process oriented data with models will be of extreme value for mastering the complexity of model driven artifacts explosion.

So far, in the current version of MARTE, the only help to address these kinds of concerns are the concepts of "mode" and "configuration". An enhanced version of MARTE should help practitioners to link models with specific phases of the development process followed. General categories described in SPEM [20] may be linked to models so that stakeholders can select and view the appropriate information at the appropriate time.

G. Methodological aspects

The ideas in this section are by far the more controversial, but we consider important to state them as a starting point for the discussion that will be needed before issuing the RFP.

One of the main criticisms of MARTE was its large size and the various ways in which a concept can be mapped into UML elements. These characteristics of MARTE responded to the various dimensions in which UML (and its profiles) can be used along the specification, development and analysis processes. A way to minimize potential inconsistencies, clarify mappings, and eventually fix the semantics of the language and its elements is by defining very precise and decoupled modeling scopes and associating to them exclusive modeling elements that may conform complementing dialects of MARTE.

Then, if the issuing companies agree, proposals might be forced to provide the needed dialects in the form of inter-related domain specific languages. All of them would be supported by a complete common meta-model whose semantics must be fixed in accordance to the UML meta-model and the executable UML related specifications. The idea here is similar to promote as normative a coherent version of all the domain models of the current version of MARTE.

The normative or non-normative quality of this meta-model needs to be discussed and clearly specified in the requirements of the RFP. Alternatively, stricter conformance cases (see clause 6 or MARTE) may be used instead to define the dialects able to be used in that strict normative way. The contents of each conformance case would need to be very clearly specified and promoted along the spec through the examples and the model libraries provided in it.

III. ELEMENTS IN A ROADMAP FOR MARTE++

By the time of writing this document, a new task force to complete the very basic issues already posted for MARTE 1.1 has been recently re-chartered at the OMG. Also an initial workshop has been held at the June 2017 OMG technical meeting to discuss potential content for an RFP and promote this effort to the interested parties in the standardization process.

If an RFP is issued by the OMG along 2017, it is possible to have an initial submission by the end of 2019. Additional efforts will be needed to have a revised submission by the end of 2020.

As it is usual in these kind of standardization efforts the contributions of the community are essential to get it on time and in shape to be of value for all stakeholders involved, scholars, practitioners, tool vendors, and regulators.

IV. CONCLUSIONS

In this short paper we have provided a brief review of analysis, design and specification concerns around the model

driven development of real-time and embedded systems. These concerns are to be discussed in a wider way in order to map them into concrete requirements that can be included in a Request for Proposals of the OMG for a new version of the MARTE standard. The main elements to consider are enhancements over its adequacy for the new challenges that CPSoS and IoT post, but also a reduction of the learning and adoption effort is a must for the real-time and embedded systems communities. On the one hand, the various efforts reviewed indicate that there are enough experiences in the community that can contribute to enhance significantly the suitability of MARTE along the software life cycle. On the other hand, its usability and better exploitation, either as a UML profile, or as a collection of well-structured domain specific languages, is an open question, which also demands some more reflection from practitioners and tool vendors.

In the near future, the real-time and embedded systems broad community have the chance to reinforce its modeling capacities with UML. The OMG is a very open and dynamic standardization body. Its policies and procedures allows any interested party to work in the aspects each feels more closely concerned, and contribute to the common understanding of the community. Being MARTE the more elaborated and larger effort defined so far by the community, working to consolidate it as the real-time and embedded system base language, is an appealing work to which all parties involved, stakeholders, practitioners, tool vendors, and regulators, are invited.

The acceptance of the needed extensions to MARTE requires a consensus based on actual limitations found in the modeling, analysis and design of real-time and embedded systems by industry. The MegaM@RT2 ECSEL project consortium will provide an initial group of companies with which these limitations can be discussed and the best ways to overcome them will be selected. This paper will serve as an initial proposal from which such technical discussion can start. As future work we consider the reviewers suggestion to extend the discussion to all inceptors and users of the current version of the standard in order to make a wider collective reflection, and prepare a more elaborated list of requirements.

ACKNOWLEDGEMENTS

This work receives funding from the Spanish Government under grant number TIN2014-56158-C4-2-P (M2C2), and from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737494 (MegaM@RT2). This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and Sweden, France, Spain, Italy, Finland, Czech Republic. We thank the anonymous reviewers for their insights and proposals of improvements

REFERENCES

[1] Object Management Group, "UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded systems," 2011 OMG Document, v1.1 formal/2011-06-02. <http://www.omg.org/spec/MARTE/1.1/PDF>

[2] B. Selic and S. Gerard, "Modeling and Analysis of real-time and embedded systems with UML and MARTE: developing cyber-physical systems. ISBN 879-0-12-416619-6, Elsevier, 2014.

[3] Object Management Group, "OMG Technology Adoption Process: Request for Proposals". <http://www.omg.org/gettingstarted/process2-RFP.htm>

[4] Object Management Group, UML Profile for MARTE 1.1 Revision Task Force, Open Issues. <http://issues.omg.org/issues/lists/marte-rtf>

[5] F. Mallet, E. Villar, F. Herrera, "MARTE for CPS and CPSoS", in S. Nakajima, J.P. Talpin, M. Toyoshima and H. Yu (Eds.): "Cyber-Physical System Design from an Architecture Analysis Viewpoint: Communications of NII Shonan Meetings", Springer, pp.81-108, doi=10.1007/978-981-10-4436-6. 2017-05

[6] AUTOSAR - Spec. of Timing Extensions, AUTOSAR Std. 4.2.2, 2014.

[7] Alan Burns and Rob Davis. Mixed Criticality Systems - A Review. 9th edition. <http://www-users.cs.york.ac.uk/~burns/review.pdf>

[8] Irv Badr, "McProcessor". <http://issues.omg.org/issues/lists/marte-rtf?view=ALL#issue-27300>

[9] F. Herrera, J. Medina, E. Villar, "Modeling Hardware/Software Embedded Systems with UML/MARTE: A Single-Source Design approach", in S. Ha and J. Teich (Eds): "Handbook of Hardware/Software Codesign", Springer, 2018..

[10] J. Medina. Perspectives on the Analysis and Design of Real-Time Distributed Systems with UML-based standard modelling languages. Workshop on Real-time systems: the past, the present, and the future, York (UK), CreateSpace Independent Publishing Platform, pp. 87-104, March 2013 ISBN: 978-1482707786.

[11] Antonio Cicchetti, Federico Cocozzi, Silvia Mazzini, Stefano Puri, Marco Panunzio, Alessandro Zovi, Tullio Vardanega. "CHESS: a model-driven engineering tool environment for aiding the development of complex industrial systems". ASE 2012: 362-365.

[12] J. Medina, and A. Pérez, "High Level Modeling for Real-Time Applications with UML & MARTE", 25th Euromicro Conference on Real-Time Systems (ECRTS'13) WiP Session, pp. 13-16, July 2013.

[13] Object Management Group, "Precise Semantics of UML Composite Structures (PSCS)", 2015 OMG Document, v1.0 formal/2015-10-02.

[14] Object Management Group, "Precise Semantics of UML State Machines (PSSM)", Version 1.0 - Beta1, 2017 OMG Document ptc/17-04-04

[15] E. Ebeid, J. Medina, D. Quaglia and F. Fummi. "Extensions to the UML Profile for MARTE for Distributed Embedded Systems". The 2015 Conference on Forum on specification & Design Languages, FDL 2015, Barcelona (Spain), September 2015.

[16] S. Bernardi, J. Merseguer and D. Petriu, "Model-Driven Dependability Assessment of Software Systems", ISBN: 978-3-642-39511-6, Springer, 2013

[17] K. Triantafyllidis, "Design Space Exploration for Component-Based Real-Time Distributed Systems". PhD Thesis. Technische Universiteit Eindhoven, 2017. Proefschrift. - ISBN 978-94-629-5596-7 NUR 959

[18] F. Herrera, H. Posadas, P. Peñil, E. Villar, F. Ferrero (GMV), R. Valencia (GMV), G. Palermo: "The COMPLEX methodology for UML/MARTE modeling and design-space exploration of embedded systems", Journal of Systems Architecture, V.60, N.1, Elsevier, pp.55-78. 2014.

[19] H. Posadas, P. Peñil, A. Nicolás, E. Villar: "Automatic synthesis of embedded SW for evaluating physical implementation alternatives from UML/MARTE models supporting memory space separation", Microelectronics Journal, V.45, I.10, pp.1281-1291, 2014

[20] Object Management Group. "Software & Systems Process Engineering Meta-Model Specification Version 2.0". OMG Document Number: formal/2008-04-01