

MARTE, THE UML standard extension for real-time and embedded systems

Sébastien Gérard ([sebastien\[dot\]gerard\[at\]cea\[dot\]fr](mailto:sebastien[dot]gerard[at]cea[dot]fr)) – CEA LIST, Laboratory of Model Driven Engineering for Embedded Systems (LISE), Boîte courrier 65, Gif sur Yvette Cedex, F-91191 France.

François Terrier ([francois\[dot\]terrier\[at\]cea\[dot\]fr](mailto:francois[dot]terrier[at]cea[dot]fr)) – CEA LIST, Laboratory of Model Driven Engineering for Embedded Systems (LISE), Boîte courrier 65 , Gif sur Yvette Cedex, F-91191 France.

Bran Selic ([selic\[at\]acm\[dot\]org](mailto:selic[at]acm[dot]org)) – Malina Software Corp., Nepean, Ontario, Canada

Pierre Boulet ([pierre\[dot\]boulet\[at\]lifl\[dot\]fr](mailto:pierre[dot]boulet[at]lifl[dot]fr)) – LIFL, Université des Sciences et Technologies de Lille, Cité scientifique, 59655 Villeneuve d'Ascq, France

Introduction

The Object Management Group (OMG, www.omg.org) is one of the principal international organizations promoting standards supporting the usage of model-based software and systems development. The Unified Modeling Language standard (UML [1]) is probably the most representative of these and has had significant success in the software industry as well as in other domains such as IT and financial systems. UML is now the most widespread language used for modeling in both industry and academia. It was designed as both a general-purpose modeling language as well as a foundation for different domain-specific languages, using its profile mechanism. The latter capability allows the general concepts of UML to be specialized for a specific domain or application. The field of real-time embedded software systems is one such domain for which extensions to UML are required to provide more precise expression of domain-specific phenomena (e.g., mutual exclusion mechanisms, concurrency, deadlines, and the like).

The OMG already had issued a UML profile for this purpose, called the *UML profile for Schedulability, Performance and Time* (SPT, [2]). It provided concepts for dealing with model-based schedulability analysis, focused primarily on rate monotonic analysis, and also concepts for model-based performance analysis based on queuing theory. In addition, SPT also provided a framework for representing time and time-related mechanisms. However, practical experience with SPT revealed shortcomings within the profile in terms of its expressive power and flexibility. For example, it was necessary to support the design of both hardware and software aspects of embedded systems and a more extensive support for schedulability and performance analysis, encompassing additional techniques such as hierarchical scheduling. Furthermore, when the new version of UML, UML 2 was adopted, it became necessary to upgrade the SPT profile. Consequently, a new Request For Proposals (RFP) was issued by the OMG seeking a new UML profile for real-time and embedded systems, named *MARTE (Modeling and Analysis of Real-Time and Embedded systems*, [4]). The intent was to address the above issues as well as to provide alignment with the UML profile for Quality of Service and Fault Tolerance (QoS & FT, [3]), which enables specification of not only real-time constraints but also other embedded systems characteristics, such as memory capacity and power consumption. MARTE was also required to support modeling and analysis of component-based architectures, as well as a variety of different computational paradigms (asynchronous, synchronous, and timed).

To respond to this RFP, a team of OMG members decided to develop a joint submission. This consortium, called the ProMARTE consortium, consisted of the following enterprises: Alcatel, ARTISAN Software Tools, Carleton University, CEA LIST, ESEO, ENSIETA, France Telecom, International Business Machines, INRIA, INSA from Lyon, Lockheed Martin, MathWorks, Mentor Graphics Corporation, NoMagic, the Software Engineering Institute (Carnegie-Mellon University), Softeam, Telelogic AB, Thales, Tri-Pacific Software, and Universidad de Cantabria. The resulting

submission was voted on and accepted by the OMG in June 2007 [5] as a “Beta Specification”.

As prescribed by the OMG's Policies and Procedures manual (P&P, [6]), following adoption, a so-called Finalization Task Force (FTF) was instituted to prepare the new specification for its formal issue as an official OMG technology recommendation. The FTF's working period is about 18 months and its first phase (around 6 months) is a comment-gathering phase during which feedback from the broader user and vendor communities is collected. Of particular significance is input from commercial and other tool vendors intending to support the new specifications in their products. The second phase is then dedicated to solving the key issues identified in the initial phase resulting in a Beta 2 version of the specification. This version is first screened by the OMG's architecture board and, if deemed acceptable, is submitted to the OMG's board of the directors for final approval. At that point, the specification becomes formally available in its version 1.0. In the case of MARTE it is expected that this will be available by the first quarter of 2009.

The Structure and Content of MARTE

As noted, MARTE is a UML profile intended for model-based development of real-time and embedded systems. It consists of a set of domain-specific extensions (i.e., specializations) of appropriate general UML concepts providing UML modelers with first-class language constructs for modeling real-time and embedded (RTE) applications. Many of these extensions pertain to so-called *non-functional* aspects of RTE¹ applications. Non-functional concerns of an application can be classified into two categories, quantitative and qualitative aspects and they may be available at different levels of abstraction. Finally, they may be defined to support modeling, analysis, or both.

In order to satisfy all these requirements, MARTE is structured as a hierarchy of (sub-)profiles, as depicted in the UML package diagram in figure 1. It has four main parts.

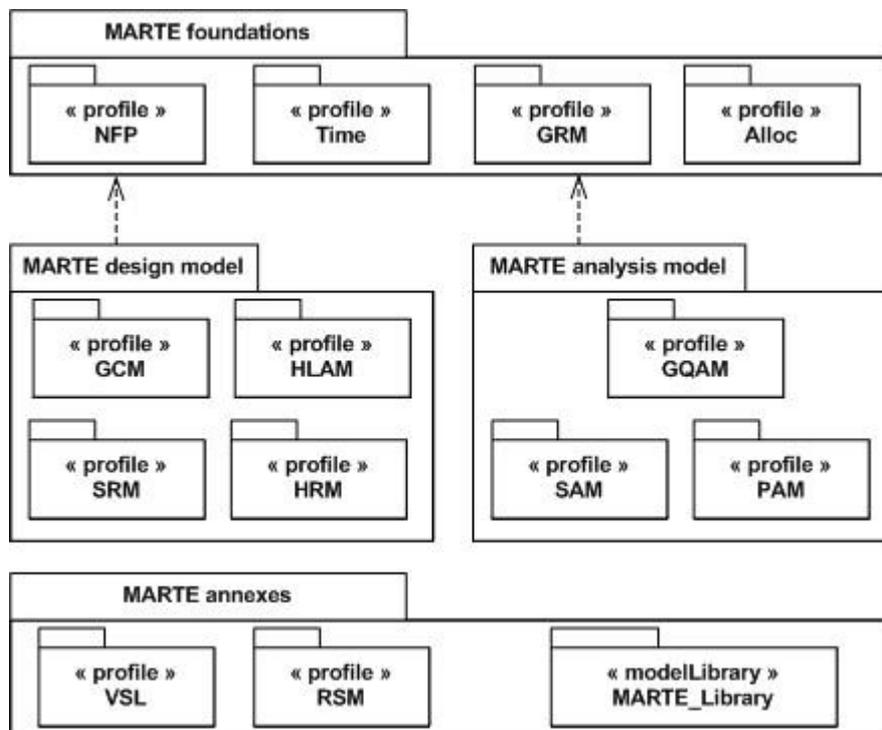


Figure 1: MARTE's Architecture Description

¹ We chose this term reluctantly, mainly because it has become very widespread. Unfortunately, it is a problematic term, since its interpretation is open-ended, covering aspects which are beyond what is intended here. Worse yet, the negative “non” prefix gives it a second-order connotation such that, in many cases, these concerns are only addressed *after* the “functional” concerns have been addressed – a situation that sometimes leads to poorly engineered systems, which in the case of RTE systems often implies failure.

The top package, which is the foundation on which the rest of MARTE is constructed, consists of four basic sets of UML extensions (sub-profiles):

- *Non-functional properties (NFP)* – This sub-profile provides modeling constructs for declaring, qualifying, and applying semantically well-formed non-functional aspects of UML models. It is complemented by the *Value Specification Language (VSL)*, which is a textual language for specifying algebraic expressions. The NFP sub-profile supports the declaration of non-functional properties as UML data types, whereas VSL is used to specify the *values* of those types and their potential functional relationships.
- *Time* – This consists of concepts for defining time in applications, and also for manipulating the underlying time representation. The Time extension defined in MARTE provides support for three qualitatively different models of time: *chronometric*, *logical*, and *synchronous*.
- *Resource (GRM)* – One important requirement with regards to RTE system modeling is to represent the set of resources underlying an application and also how the system uses them. The *GRM (Generic Resource Modeling)* sub-profile consists of an ontology of resources enabling modeling of common computing platforms (i.e., a set of resources on top of which an application may be allocated to be computed), and high level concepts for specifying resource usage. The level of abstraction used here is at a general system level.
- *Allocation modeling (Alloc)* – This sub-profile of the foundational layer provides a set of general concepts pertaining to allocation of functionality to entities responsible for its realization. It may be either time-related allocation (i.e., scheduling) or space allocation. It also tackles the more abstract issue of refinement between models at different levels of abstraction. Note that non-functional characteristics may be attached to an allocation description (e.g., when specifying the allocation of a function to a given execution engine, it is possible to specify its worst case execution time).

Starting from these foundational concepts, MARTE is then split into two different categories of extensions: One (denoted in figure 1 as the “MARTE design model”) is dedicated to supporting the activities of the left branch of the classical “V” development cycle (i.e., model-based design), whereas the other (denoted by the “MARTE analysis model” package), provides support for model-based analysis of RTE applications (i.e. more devoted to the validation and optimization activities).

Model-based design of RTE systems with MARTE proceeds mostly in a declarative way. Hence, MARTE users may annotate their models with real-time or embedded concerns using the extensions defined within the *HLAM (High-Level Application Modeling)* sub-profile. For instance, concurrent computing units with real-time features may be denoted using an extension called «rtUnit» and, by giving specific values to its properties, they can also indicate what is the model of computation for the concurrent unit. Note also that MARTE enables component-based system (either software or hardware) engineering through its specific sub-profile, *GCM (Generic Component Model)*. GCM supports both message- and data-based communication schemes between components. In addition, MARTE also defines very refined concepts that enable users to describe its computing platforms (either software or hardware) in a very detailed and precise manner [7, 8, 9]. As examples, MARTE includes in its annexes facilities for modeling OSEK, ARINC or POSIX compliant software computing platforms. Finally, to deal effectively with the increasing degrees of parallelism available on chips, MARTE provides in an annex the *RSM (Repetitive Structure Modeling)* sub-profile that enables a compact representation of multidimensional regular structures.

Model-based analysis using MARTE is done using mainly the extensions defined either in the *Generic Quantitative Analysis Modeling* profile (*GQAM*), or using one of its two refinements, dedicated respectively to schedulability analysis and performance analysis. The annotation mechanism used in MARTE to support model-based analyses uses UML stereotypes. These typically map the UML model elements of the application into corresponding analysis domain concepts, and also allow specification of values for properties which are needed to carry out the

analyses.

Finally, MARTE fosters model processing in the way that it enables adding semantics to a given UML model (e.g., for code production or quantitative analysis purposes). In this context, one generic usage of MARTE enables the model-processing schema described in Figure 2. Note that this process can be highly automated, which, in some cases, can even eliminate the need for analysis domain experts which are often difficult to obtain.

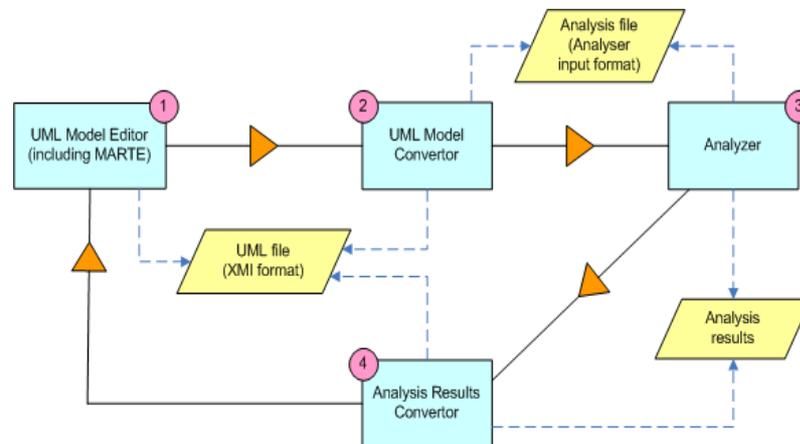


Figure 2: Schema of the MARTE model-processing framework

Future

At the time of this writing, MARTE is available in its Beta 2 version on the OMG web site (www.omg.org). In March 2009, it should be available in its version 1.0 after which the OMG will start successive minor revision cycles (lasting 18 months on the average), each one representing a new “dot” release. The purpose of a revision task force is to recommend changes that clarify complex or confusing text in the spec, or make revisions that correct errors. Note, however, that these revisions cannot add new capabilities or enhancements to the profile. For that, the OMG needs to issue a new RFP.

MARTE has already been applied extensively in practice both by industry and academia as indicated by the list of ongoing project that put MARTE in the center of their concerns (cf. the OMG web site dedicate to MARTE, www.omgmarTE.org).

Links and References

- [1] OMG, UML Version v2.1.2, formal/07-02-05, <http://www.omg.org/spec/UML/2.1.2/>.
- [2] OMG, UML Profile for Schedulability, Performance, and Time, v1.1, formal/05-01-02, <http://www.omg.org/technology/documents/formal/schedulability.htm>.
- [3] OMG, UML Profile for Modeling QoS and FT Characteristics and Mechanisms, v1.0, formal/06-05-02, <http://www.omg.org/spec/QFTP/1.1/>.
- [4] OMG, UML Profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE) RFP, realtime/05-02-06.
- [5] OMG, UML Profile for MARTE, Beta 1, ptc/07-08-04, <http://www.omg.org/cgi-bin/doc?ptc/2007-08-04>.
- [6] OMG, OMG Policies and Procedures, Version 2.7, pp/08-06-01, <http://www.omg.org/cgi-bin/doc?pp>
- [7] F. Thomas, S. Gérard, J. Delatour and Francois Terrier, Software Real-Time Resource Modeling, In Proceedings of the International Conference Forum on Specification and

- Design Languages (FDL) 2007, Barcelona, Spain, September 2007.
- [8] S. Taha, A. Radermacher, S. Gerard, and J.-L. Dekeyzer, An Open Framework for Hardware Detailed Modeling, In IEEE Proceedings of SIES'2007, ISBN 1-4244-0840-7, Lisbon, Portugal, July 2007.
 - [9] S. Taha, A. Radermacher, S. Gerard, and J.-L. Dekeyzer, MARTE: UML-based Hardware Design from Modeling to Simulation, In Proceedings of the International Conference Forum on Specification and Design Languages (FDL) 2007, Barcelona, Spain, September 2007.