Boosting productivity in embedded-systems development

MARTES
(ITEA ~ 04006)

By: Klaus Kronlöf, Nokia Research Center, Finland

The embedded systems industry faces major challenges in improving productivity as systems complexity increases. The ITEA MARTES project exploits model-based methodology with a combination of the Unified Modelling Language (UML) and SystemC hardware description languages that makes it possible to verify architecture early in the design cycle, speeding development of real-time embedded systems and markedly reducing time to market. Advantages include improved interoperability of software engineering tools. The validity of the approach has been demonstrated in a series of industrial case studies.

Digital convergence leads to new products created by combining and integrating existing and new technologies in innovative ways. However, such developments often leave embedded systems developers struggling with unprecedented complexity and scalability. Traditional programming craftsmanship cannot deal with all these problems, so a real engineering methodology is needed.

MARTES set out to improve productivity and reduce the complexity of embedded systems development. It defined, constructed, validated and deployed a model-based methodology that separates application development and platform technology, together with an interoperable toolset for real-time embedded systems (RTES) development.

The resulting methodology is applicable in principle to the development of any kind of RTES. It focuses on early phases, such as requirement specification and architecture definition. The methodology was evaluated by industrial partners in the domains of mobile communications, telecommunications, consumer multimedia, avionics command/control and displays, and space. These industrial case studies showed encouraging results.

Driven by systems companies and tool providers

MARTES was driven by the needs of systems companies and tool providers, both well represented in the large project consortium:

- **Systems companies** urgently need the new methodology and supporting tools for efficient development of embedded systems products targeting increasingly diverse markets. The co-modelling approach enables analysis and early verification of design architecture. This capability is crucial to maintaining and advancing the leading position of European companies; and

- **Tool providers** require a consistent, widely accepted methodology and tool interoperability for the creation of a large market for systems-level design tools. UML and model-driven architecture (MDA) have already gained widespread acceptance in the information technology industry, and these concepts can form a basis for an industry-standard, embedded-systems design methodology. This opens up a market for a new category of UML-based tools – and, thanks to tool interoperability, increases the market for many existing tools.

The success of MARTES gives tool vendors the opportunity to enlarge market coverage by supporting embedded-systems development more extensively and by realising dedicated tools for model-based engineering, thanks to a close partnership with the main actors in this domain.

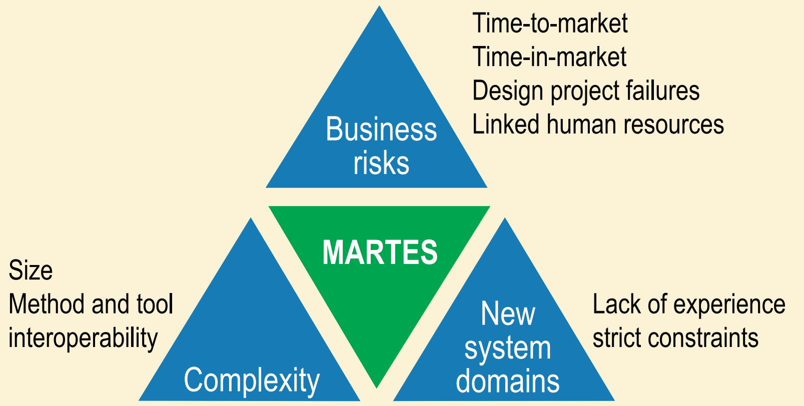

Figure 1: Embedded systems industry product development challenges

Use of UML expanding significantly

UML is a general purpose modelling language already widely used in software engineering, while SystemC is a popular C++-based hardware modelling language for transaction-level simulations. MARTES set out to combine the strengths of both languages to simplify modelling and help increase the abstraction level, especially on the software side, where SystemC alone does not have extensive support.

In recent years, use of UML has expanded significantly in the embedded systems industry. While software designers were the only users in the past, there are now new users in systems-architecture creation and communication-protocol engineering.

The notations of UML 2.0 seem appropriate to represent many of the relevant architectural views for systems architects, but the analytical capabilities of available UML tools are limited. In fact, there are hardly any UML-based architectural/performance analysis tools available – offering a major market opportunity.

SystemC 2.1 was ratified as the IEEE 1666 standard in December 2005 and is being adopted for hardware modelling at higher levels of abstraction – the only real competitor seems to be System-Verilog for existing Verilog users. SystemC will probably be widely used for hardware-oriented architectural models – i.e. for modelling platform resources, such as processors.

This kind of modelling is largely complementary to use of UML for more end-user-oriented architectural modelling. However, there is an obvious need for interoperability. Proper interworking of UML and SystemC increases the attractiveness and market for both UML and SystemC tools as well as commercial off-the-shelf (COTS) SystemC models.

Exploiting a common meta model

A major achievement of MARTES was to bring together the views from the many different industries in the consortium into a common understanding on model-based RTES development. At the core of the methodology is a meta model, which forms the basis of a new way to use UML and SystemC for RTES development and of the MARTES UML profile.

In practice, software applications, their control and scheduling are modelled with UML 2.0, while the hardware architecture model is written with the SystemC. In architectural exploration the application model represents the computational workload while the hardware architecture model represents the execution resources. The combined model can be simulated to monitor the timing of execution and resource use. The interface between the workload and hardware models is also described in UML to enable simulation of the modelled system.

The strength of this approach is that it enables architectural exploration and dealing with non-functional properties, such as real-time performance and power, better than any previous UML approach. In addition, the MARTES meta model enables semantically sound integration of tools for architecture development, analysis and deployment.

Separating application and platform development

A fundamental idea of the MARTES common core meta model is to separate the concerns of application and execution platform development. This brings three benefits:

1. **Complexity management** – the applications and platform aspects of the system can be developed and analysed separately;

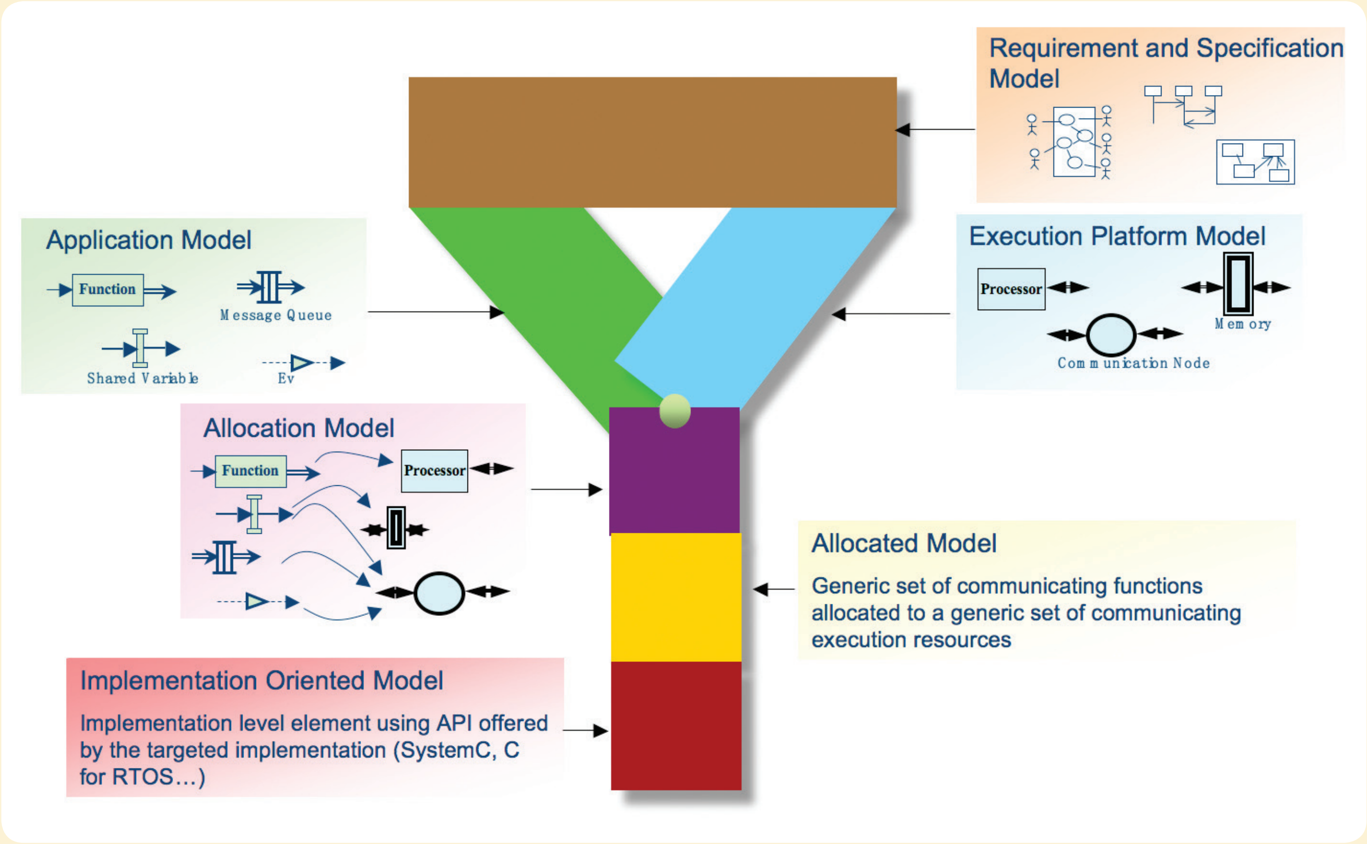

Figure 2: Models of the MARTES methodology

2. **Architectural exploration** – the application model is mapped onto the platform model explicitly, and several alternatives can be tried to find the best solution; and
3. **Re-use** – application and the platform models are both self-contained re-usable design artefacts.

The MARTES UML profile is based on this meta model and is divided in the same way into application, platform and allocation models with associated stereotypes.

A methodology was defined for model transformations, based on the core meta model. MARTES methodology also includes the development process aspect. It defines a tailorable model-driven engineering process framework that supports traceability and conforms to standardisation requirements.

All the tools in the MARTES tool integration strategy share a common technical space, enabling tool interoperability. Each domain-specific tool has its own extended technical space that is partly mapped to this common space, which is defined by the MARTES meta model using EMF Ecore technology[1].

Aiming for consensus on interoperability

The primary motivation for systems companies to join this ITEA project was obviously the improvement in their product development practices. This should lead to more cost-efficient and better quality products. Likewise, tool vendors aimed to improve their product offerings and so attract new customers. A number of commercial and in-house tools have been implemented. UML tools use the MARTES profile, while other tools are integrated using model transformations and the meta model.

However, the impact of the project is expected to be much wider. The ambition is to achieve a consensus in the industry at large on the common meta-model-based interoperability. Several MARTES partners have participated actively in the ProMARTE consortium[2], supplying project results as contributions to the definition of the OMG UML MARTE profile[3].

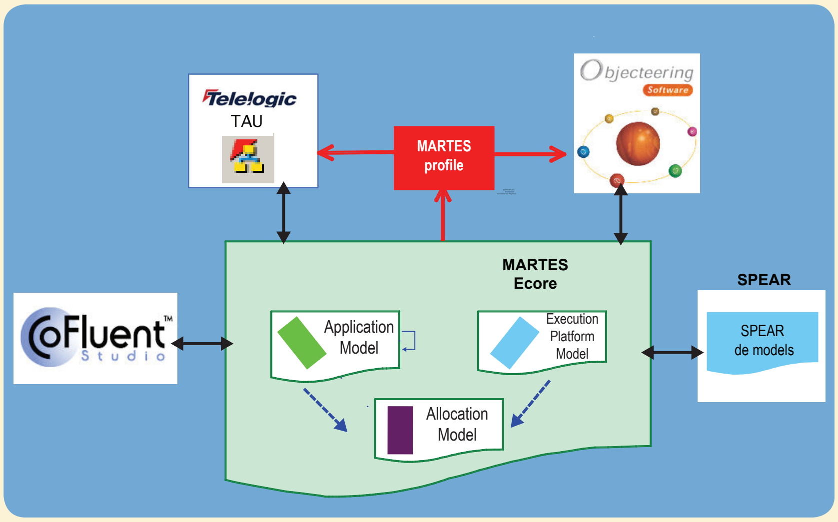

Figure 3: MARTES tool interoperability framework

The benefits of industry-wide meta-model adoption would be significant in terms of solving product development challenges in embedded systems through model-based thinking by:

- Improving efficiency in joint product development;
- Enabling interoperable models and thereby easing use of COTS, intellectual property (IP) components and subsystems; and
- Easing integration of the design environment and so enabling use of best-in-class tools from multiple vendors. This opens up a market for small tool vendors offering highly focused tools.

References

[1] *http://www.eclipse.org/modeling/emf/*

[2] *http://www.promarte.org*

[3] *http://www.omg.org/cgi-bin/doc?ptc/2007-08-04*