The MegaM@Rt2 ECSEL Project
MegaModelling at Runtime – Scalable Model-based Framework for Continuous Development and Runtime Validation of Complex Systems

Wasif Afzal, Hugo Bruneliere

Mälardalen University, Sweden & IMT Atlantique - LS2N (CNRS) - ARMINES, France

Davide Di Ruscio, Andrey Sadovykh

University of L'Aquila, Italy & Softeam, France

Silvia Mazzini, Eric Cariou

Intecs, Italy & Université de Pau et des Pays de l'Adour, France

Dragos Truscan, Jordi Cabot, Abel Gómez

Åbo Akademi University, Finland & ICREA, Spain & Universitat Oberta de Catalunya, Spain

Yosu Gorroñogoitia, Luigi Pomante, Pavel Smrz

ATOS, Spain & University of L'Aquila, Italy & Brno University of Tech., Czech Republic

Abstract

A major challenge for the European electronic industry is to enhance productivity by ensuring quality of development, integration and maintenance while reducing the associated costs. Model-Driven Engineering (MDE) principles and techniques have already shown promising capabilities, but they still need to scale up to support real-world scenarios implied by the full deployment and use of complex electronic components and systems. Moreover, maintaining efficient

*Corresponding author

Email addresses: wasif.afzal@mdh.se & hugo.bruneliere@imt-atlantique.fr (Wasif Afzal, Hugo Bruneliere), davide.diruscio@univaq.it & andrey.sadovykh@softeam.fr (Davide Di Ruscio, Andrey Sadovykh), silvia.mazzini@intecs.it & eric.cariou@univ-pau.fr (Silvia Mazzini, Eric Cariou), dragos.truscan@abo.fi & jordi.cabot@icrea.cat & agomezlla@uoc.edu (Dragos Truscan, Jordi Cabot, Abel Gómez), jesus.gorronogoitia@atos.net & luigi.pomante@univaq.it & smrz@fit.vutbr.cz (Yosu Gorroñogoitia, Luigi Pomante, Pavel Smrz)

traceability, integration, and communication between two fundamental system life cycle phases (design time and runtime) is another challenge requiring the scalability of MDE. This paper presents an overview of the ECSEL[1] project entitled "MegaModelling at runtime – Scalable model-based framework for continuous development and runtime validation of complex systems" (MegaM@Rt2), whose aim is to address the above mentioned challenges facing MDE. Driven by both large and small industrial enterprises, with the support of research partners and technology providers, MegaM@Rt2 aims to deliver a framework of tools and methods for: 1) system engineering/design and continuous development, 2) related runtime analysis and 3) global models and traceability management. Diverse industrial use cases (covering strategic domains such as aeronautics, railway, construction and telecommunications) will integrate and demonstrate the validity of the MegaM@Rt2 solution. This paper provides an overview of the MegaM@Rt2 project with respect to its approach, mission, objectives as well as to its implementation details. It further introduces the consortium as well as describes the work packages and few already produced deliverables.

Keywords: `Model-Driven Engineering`, `Design Time`, `Runtime`, `Megamodelling`

---

1. Introduction

In the global context, the European electronic industry faces stiff competition. Electronic systems are becoming more and more complex and software intensive [1], which calls for novel engineering practices to tackle advances in productivity and quality of these, now, cyber-physical systems [2].

Model-Driven Engineering (MDE) refers to a system development methodology where abstractions—or models—are systematically used along the process [3]. MDE promises many potential benefits (e.g., gains in productivity, portability, maintainability or interoperability) and several studies have been

[1] `http://www.ecsel-ju.eu/web/index.php`

conducted to support these claims with empirical data [4–9]. Moreover, in the last years, the technological ecosystem around MDE has flourished, providing developers with a plethora of tools to support modeling tasks, ranging from model management solutions to model transformation and code-generation engines. However, these technologies need to be further developed to scale for real-life industrial projects and provide advantages at runtime. The ultimate objective of enhancing productivity by ensuring quality of development, integration and maintenance while reducing the associated costs can be achieved by the use of techniques that integrate design and runtime aspects within system engineering methods incorporating existing engineering practices [10]. Industrial scale models, which are usually multi-disciplinary, multi-teams, combine several product lines and typically include strong system quality requirements, can be exploited at runtime by advanced tracing and monitoring. Thus, achieving a continuous system engineering cycle between design and runtime, ensuring the quality of the running system and getting valuable feedback from it that can be used to boost the productivity and provide lessons-learnt for future generations of products [11].

A major challenge in the Model-Driven Engineering of critical software systems is the integration of design and runtime aspects. The system behavior at runtime has to be matched with the design in order to fully understand critical situations, failures in design, and deviations from requirements. Many methods and tools exist for tracing the execution and performing measurements of runtime properties (see e.g. [12][13]). However, most of these methods do not allow the integration with system models – the most suitable level for system engineers for analysis and decision-making.

The MegaM@Rt2 (MegaModelling at Runtime) proposal was submitted to the ECSEL in 2015. It received good evaluation scoring: 4.3 in Excellence, 4.6 in Impact and 4 in Implementation. The overly positive and instructive remarks motivated us to continue with MegaM@Rt2 in 2016, and a proposal was submitted for the research and innovation action in the call H2020-ECSEL-2016-RIA, by reinforcing the consortium and clearing the project details. The project officially started on April 1, 2017 and runs for 3 years.

The vision of MegaM@Rt2 is to create a scalable framework for model-based continuous development and validation of large and complex industrial systems by exploiting important features of:

- MARTE, SysML, and others, to express both system functional and non-functional properties;
- model-based verification and validation methods at design time and runtime;
- methods for model management/megamodelling;
- methods for traceability over large multi-disciplinary models;
- methods for inference of system deviations from expected behavior and affected design elements.

This article is an extension of our previous conference paper [14]. Compared to it, we have added many details on the current status of the project as well as the already produced deliverables. At the time of this submission, the project is about to enter its second year of activity. So far, industrial case study requirements and the baseline methodologies provided by the project partners have been collected and analyzed in the context of Work Package 1 (WP1). In addition, a detailed study of the state-of-the art has been performed and corresponding needs for innovation have been identified as part of Work Packages 2, 3 & 4 (WP2–4). In the upcoming phase of the project, we will perform gap analysis between the industrial needs and the baseline methodologies & tools. Moreover, we will suggest the main features of the MegaM@Rt2 framework as well as a related road map presenting how the different components will be further developed.

Section 2 outlines the mission and the objectives of MegaM@Rt2. Section 3 discusses the main concepts of the development approach proposed by the project. The potential industrial impact of MegaM@Rt2 is summarized in

Section 4. The partners of the consortium are presented in Section 5. The work packages (WPs) aiming at achieving the objectives of MegaM@Rt2 are presented in Section 6, whereas Section 7 presents a brief description of each WP along with briefly describing some already produced deliverables. Section 8 concludes the paper.

2. Project Mission and Objectives

The mission of MegaM@Rt2 is to create a framework incorporating methods and tools for continuous system engineering and validation leveraging the advantages of scalable model-based methods. This will provide benefits in significantly improved productivity, quality and predictability of large and complex industrial systems. Such a mission is realized through the following specific objectives:

- **Objective 1.** *MegaM@Rt2 continuous system engineering:* to develop scalable methods and tools for the integration of design artifacts resulting from heterogeneous engineering practices, including the modelling of functional and non-functional properties (e.g. performance, energy consumption, security and safety) based on requirements.
- **Objective 2.** *MegaM@Rt2 runtime analysis:* to develop integrated methods and tools for trace analysis based on probes injection to runtime artifacts, as well as improved monitoring in order to validate the system-level requirements.
- **Objective 3.** *MegaM@Rt2 (global) model management:* to develop scalable infrastructure for efficient handling and management of numerous, heterogeneous, and large models potentially covering several functional and non-functional aspects.
- **Objective 4.** *MegaM@Rt2 unified traceability management:* to develop holistic traceability methods and tools 1) able to link and manage models and their elements from different tools as well as 2) suitable for large

distributed cross-functional working teams and 3) allowing to integrate the feedback to the system level models.

- **Objective 5.** *MegaM@Rt2 demonstrators validation:* to develop specific demonstrators and validate MegaM@Rt2 technologies through 9 complementary industrial case studies.
- **Objective 6.** *MegaM@Rt2 market uptake:* to prepare exploitation of the MegaM@Rt2 technology through open source and commercial tools.

3. Concept and Approach

In the past, MDE principles and techniques have already shown promising capabilities that have been experimented in a context having software components relying on hardware configurations and their interactions e.g., with their underlying environment, being very often numerous, complex, heterogeneous and strongly interrelated. However, they have generally failed in terms of 1) scalability to support real-world scenarios implied by the full deployment and use of complex electronic components and systems (ECS) and 2) maintaining efficient traceability, integration and communication between two fundamental system life-time phases which are design time and runtime, notably as far as non-functional properties and their verification & validation aspects (see e.g. [15][16]) are concerned.

As a consequence, the overall idea of MegaM@Rt2 is to scale up the use of model-based techniques by offering proper methods and related tooling, interacting with both design time and runtime, as well as to validate the designed and developed approach in concrete industrial cases involving complex ECS. To this intent, MegaM@Rt2 proposes an overall model-based approach combining existing and novel techniques. A fundamental challenge notably resides in providing efficient traceability support between the two levels i.e., from design models to runtime ones and back. Moreover, modern large-scale industrial software engineering processes require thorough configuration and model governance to provide the promised productivity gains. Thus, a scalable megamodelling approach is required to manage all the involved artifacts e.g., the many different models, corresponding work flows, configurations, etc. and to better tackle their large diversity in terms of nature, number, size, complexity, etc. Verification and validation of highly configurable systems thus also takes importance (see e.g. [17]).

To cover all these topics and deal with the complete value chain, MegaM@Rt2 brings together prominent tool vendors and research organisations with state-of-the-art methods and tools to be validated in highly relevant European industrial case studies. The end users from the space, naval, railway, smart grid, smart warehouse and telecom industry domains aim to drive the project by providing real-world requirements and case studies as well as by validating and endorsing the MegaM@Rt2 results.

Figure 1 provides an overview of the MegaM@Rt2 global approach and emphasizes its key principles and concepts, relating them to the corresponding work packages (described in detail in Section 6). A set of current engineering practices based on SysML, AADL, EAST ADL, but also Matlab/Simulink, AUTOSAR and Method B or Modelica, each one producing as set of specific design models, requirement specifications and resulting software (and sometimes also hardware) artefacts, are integrated into a global system model providing a complete view of the cyber-physical system, and detailing the components, behaviour and desired quality properties of the system. These properties are then object of exhaustive continuous testing and monitoring in the runtime environment (thanks to the configuration of the target platform and the injection of probes in the software or also in the hardware [18][19][20]) to detect deviations in real-time. These deviations, plus all the traces information collected in the process, are analyzed to detect the impacted components in the integrated view of system models. When possible, automatic repairing recommendations will be provided to correct the identified issues and reconfigure or redeploy the system to start the next iteration of the continuous integration process.

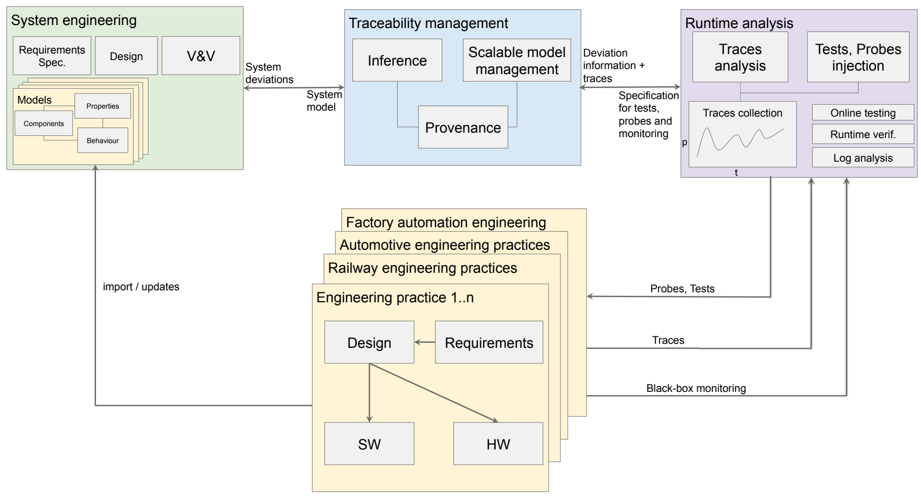

Figure 1: The MegaM@Rt2 Overall Approach.

4. Industrial Impact

The ECSEL[2] program seeks to invest in projects that strengthen the industrial competitiveness, enable economic growth and improve sustainability. Europe has a reasonably strong position in the world embedded market (30%), but this is falling as other geographies grow – some at the vanguard and others catching up. The MegaM@Rt2 consortium argues that investment in capability of the software development tools market, although only a fraction, has a very large pay-off. We have seen that the software component of the systems is increasingly more growing in importance. As the hardware becomes commoditized, the added value will rapidly shift to the software. Achieving technological and competitive superiority in software development tools will allow European firms to participate with greater dominance in the overall software market.

Specifically, MegaM@Rt2 achieves this in part through reducing development and exploitation costs and in part by allowing mastery of more complex systems. Reducing development costs and time-to-market is a competitive ad-

[2] http://www.ecsel-ju.eu/web/index.php

vantage, allowing on, one hand, greater innovation in each product and allowing faster reaction to hardware changes or new usage scenarios on the other. As the Cyber-Physical Systems' world evolves, the agility to react rapidly to new opportunities is a critical success factor for businesses. Mastering ever more complex systems allows new usage scenarios to emerge, based on optimization of greater problems or more optimized solutions for existing ones.

Improved software will allow the bigger players to better position their overall solutions and engender small businesses fulfilling niche needs for high end bespoke software. Investment in this area is timely and appropriate. The small scale and the under-developed capacity of this market segment can lead to large pay-offs in the related fields, whereas the overall embedded systems are of such a magnitude that it requires vast research investment for significant progress.

The MegaM@Rt2 objectives address several market trends in Cyber-Physical Systems:

- Increasing inclusion of advanced techniques like model-based design, development and validation.
  - MegaM@Rt2 supports this trend in the technologies provided through industrial case studies.
- Technology availability and support during extended period (e.g., up to 30 years in the railways).
  - MegaM@Rt2 open source solutions support this requirement.
- Convergent combination of multi-domains industrial practices.
  - MegaM@Rt2 supports this challenge with multi-domain case studies.
- More and more complex (structure/behaviour) connected systems.
  - With a clear support for megamodelling and system analysis at runtime, MegaM@Rt2 supports this trend.

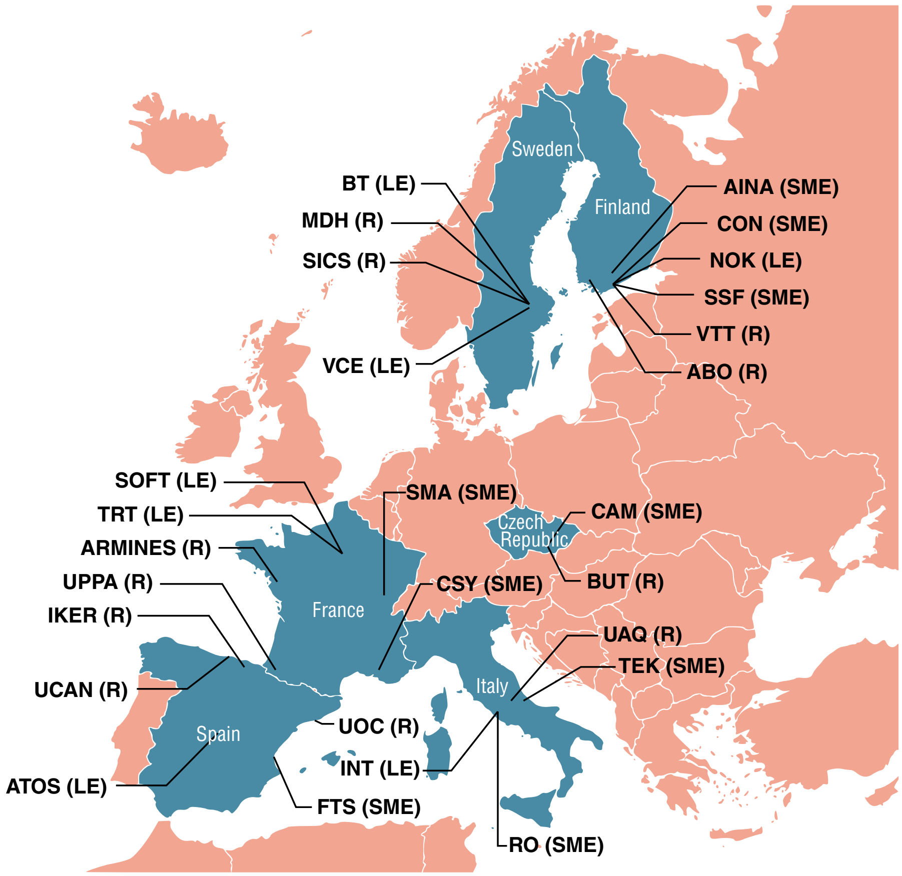

Figure 2: The MegaM@Rt2 Consortium.

5. Consortium

The MegaM@Rt2 consortium is large and is composed of partners having different complementary profiles. It brings together 27 partners coming from 6 European countries, each of which constitutes a national consortium (France, Spain, Italy, Sweden, Finland and Czech Republic). See Figure 2 (the abbreviations used in partner names are described in Sections 5.1, 5.2, 5.3).

The project consortium is strongly industry-led and consists of 7 Large Enterprises (LE) and 9 Small and Medium Enterprises (SME) accompanied by 11 universities or research and technology transfer organizations (R). An adequate

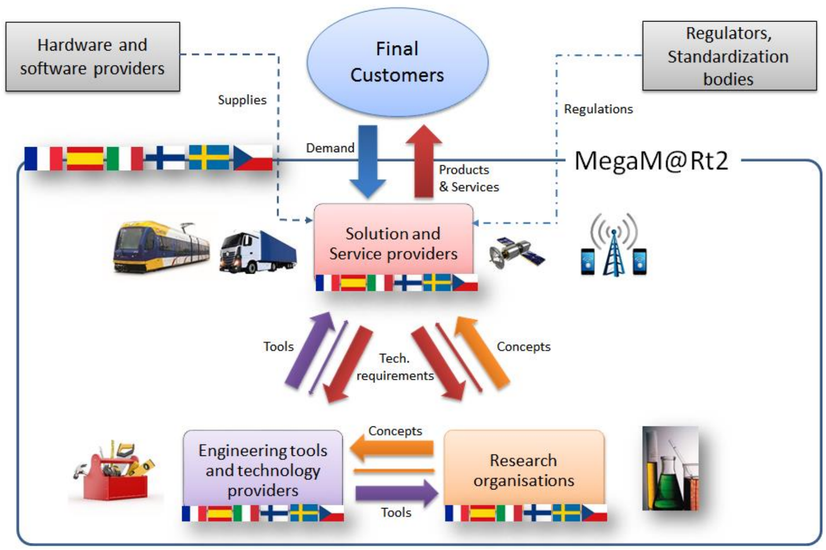

Figure 3: The MegaM@Rt2 project involves partners covering the market and technology value chains.

level of balance has been achieved by choosing SOFTEAM as a technical coordinator (a French LE with comprehensive experience in managing large research projects) while the managerial coordination is led by Mälardalen University (Sweden), which also has an extensive experience in both, participating and managing, EU projects. A suitable management strategy has been evolved by bringing together partners that know each other and have already collaborated in the past [21]. To setup the consortium, a complete value-chain has been taken into account by selecting case study owners, technology providers, and research partners (Figure 3):

- **Case study owners and end-user partners.** Providing knowledge of both end-users needs and development scenarios for complex industrial systems.
- **Technology and service providers partners.** Providing knowledge and tools in MDE, hardware and software synthesis, collaborative modelling and standardization.

- **Research partners.** Providing knowledge in megamodelling, MDE, code generation, Verification & Validation and logs analysis.

In the remaining of the section, all the members of the consortium are briefly described with respect to their role in the project.

5.1. *Case study owners and end-user partners*

Nine industrial partners will play the role of case study providers and end-users as described below.

Thalès Research & Technology – TRT (FR) provides a case study in avionics domain and will lead the validation scenarios definition. TRT has an extensive experience with MDE.

ClearSy System Engineering – CSY (FR) provides a case study in safety critical railway systems.

IKERLAN S. Coop – IKER (ES) provides a case study in smart warehouse domain and will lead the experiments with baseline technologies.

Tekne – TEK (IT) provides a case study in short-range communications domain and will lead the requirements analysis activities.

Nokia – NOK (FI) provides a case study in the telecommunications domain and will lead the case studies development activities.

Bombardier Transportation Sweden AB – BT (SE) provides a case study of their train control and management system (train/railway domain).

Volvo Construction Equipment AB – VCE (SE) provides a case study in the vehicular domain (VCE's electrical and electronic system technology platform).

Camea – CAM (CZ) provides the case study in vision-based intelligence.

AinaCom Oy – AINA (FI) will provide a case study in the communication gateway domain.

The technological domains and the applicationa areas of the case studies in the project are summarized in Table 1.

Table 1: Case studies from MegaM@Rt2 Partners.

No.	Technological Domain	Application specific
1	Avionics	Flight Management System
2	Railway	Platform Screen Doors Control
3	Smart warehouse	Deployment and Supervision of Agents
4	Short range communications	Indoor Positioning
5	Telecommunications	Base Transceiver Station
6	Transportation	Train Control and Management System
7	Automotive	Engine Control
8	ICT Services	SMS Gateway
9	Traffic monitoring	Intelligent Traffic Surveillance System

5.2. Technology and service providers partners

Eight industrial partners will play the role of technology and service providers as described below.

Softeam – SOFT (FR) will contribute with its expertise in MDE as a tool vendor for Modelio work bench and as an active member of the Object Management Group. SOFT's technical contribution will include the work on user interface generation from Interaction Flow Modeling Language (IFML) specification, code generation with "MDD+aspects" approach and scalable model management with model fragments infrastructure.

Smartesting Solutions & Services – SMA (FR) will lead the work package on runtime methods and tools. SMA's main contribution will be in online testing techniques development. SMA will contribute to baseline technologies with SmartTesting CertifyIt technology.

ATOS Spain – ATOS (ES) will lead the MegaM@Rt2 framework integration and the exploitation work package. ATOS will contribute to model simulation task force and code generation by providing development for Foundational UML (fUML) and AspectJ.

Fent Innovative Software Solutions – FTS (ES) expertise is focused on the development of execution platforms for mixed criticality systems. It is specialized in: (1) Design and development of hypervisor technology; (2) Design and development of real-time operating systems; (3) Adaptation of operating systems to be executed as a partition on top of XtratuM hypervisor. FTS will mainly be involved in Runtime work package and will provide its expertise in execution platforms.

Intecs – INT (IT) contributes to the MegaM@Rt2 framework with the CHESS model-driven, component-based methodology and tool chain for the development of high-integrity systems for different domains. INT participates in the development of the CHESS open source project delivered under Eclipse/Polasys. CHESS relies on MARTE, with focus on non functional properties modelling, analysis and correct-by-construction code generation.

Ro Technology – RO (IT) will provide advanced design, development and V&V Techniques.

Space Systems Finland Ltd. – SSF (FI) will contribute to the MegaM@Rt2 framework with the LIME toolset for runtime monitoring of the implementations and automatic test generation, which was partially funded by SSF. SSF will work on integrating the toolset to other MegaM@Rt2 tools. SSF will also participate in the application of the tools to the case studies provided by other Finnish partners. Additionally, SSF will share its extensive knowledge of verification and validation methods for safety-critical systems.

Conformiq Software Oy – CON (FI) will contribute to model-based functional test generation in all stages of software process, and to model-based test validation by functional coverage and test correctness analysis with respect to system models. Conformiqs focus is in behavioural models in contrast to e.g. purely architectural models. Conformiq will work on integrating the technology platform to other MegaM@Rt2 tools. In addition, Conformiq will participate in the deployment and application of the platform to the case studies.

Table 2: MegaM@Rt2 Research Partners.

Name	Contributions
Association pour la Recherche et le Développement des Méthodes et Processus Industriels / Institut Mines-Télécom – ARMINES (FR)	Leads activities on scalable model management & traceability.
Université de Pau et des Pays de l'Adour – UPPA (FR)	Leads activities on models' execution techniques development and contribute with the PauWare library.
Universidad de Cantabria – UCAN (ES)	Leads development of the design level verification and validation methods tools. Contributes with eSSYN tool suite featuring software synthesis technology.
Universitat Oberta de Catalunya – UOC (ES)	Leads development of scalable model-based techniques. Contributes with EMFtoCSP verification tool suite.
Universit degli Studi dell'Aquila – UAQ (IT)	Leads the traceability and provenance task force.
Åbo Akademi University – ABO (FI)	Leads the runtime verification task and contributes to all the work packages providing expertise in Aspects Oriented Modelling. Further contributes with UPAAL TRON tool suite.
Teknologian tutkimuskeskus VTT Oy – VTT (FI)	Leads development activities in logs analysis with machine learning and data mining technologies.
RISE SICS Västerås AB Västeras – SICS (SE)	Contributes in runtime verification and validation methods, their implications and required support from higher modelling levels.
Mälardalen University – MDH (SE)	Contributes in verification and validation at design-time, verification and testing at run-time, integration of megamodelling and traceability within the overall tool chain.
Brno University of Technology – BUT (CZ)	Contributes in runtime model optimization and validation through classification and scheduling methods from historical performance data.

5.3. Research partners

Ten partners will drive the research activities of the consortium. Their names and contributions in the project are summarized in Table 2.

6. Work Packages

The main expected result of MegaM@Rt2 is a practical framework incorporating methods and tools for continuous system engineering and validation. As introduced earlier, its overall goal is to leverage the advantages of scalable model-based methods to provide significantly improved productivity, quality and predictability of large and complex industrial systems. This framework will be composed of three main tool sets for 1) system engineering/design & continuous development, 2) related runtime analysis, and 3) global model & traceability management (respectively). As a consequence, we have organized the project around the research work and realization of these tool sets. Their integration and actual application onto a set of concrete use cases, covering different industrial domains, is also a central aspect of the project.

To reflect these principles, the project has been organized in 7 complementary work packages (WPs):

- WP1. Case Study Requirements Analysis & Architecture Specification;
- WP2. MegaM@Rt2 System Engineering;
- WP3. MegaM@Rt2 Runtime Analysis;
- WP4. MegaM@Rt2 Global Model & Traceability Management;
- WP5. Integration, Case Study Development & Evaluation;
- WP6. Dissemination and Exploitation;
- WP7. Management.

The work to be realized in the project is strongly requirements-driven. These requirements are extracted from the use cases as part of WP1, by exploiting the collaboration among the use case providers (mainly large industrial companies) and the technical providers (composed of both service/product companies and experienced researchers from academia). WP1 is also in charge of defining the overall architecture (conceptual and technical) of the MegaM@Rt2 solution. Most of the research and development effort is concentrated in WP2, WP3 and WP4, which aim at providing the three tool sets previously mentioned. Within WP5, these technical results will be then integrated together, applied on the use cases and finally evaluated for further improvement. The work in the project will follow an iterative and incremental approach divided into three consecutive phases. In the first phase, we will specify the requirements, validation scenarios, global architecture and roadmap. In addition, case study partners will experiment with baseline technologies while technology providers will develop the first set of prototypes. In the second phase, we will consolidate these prototypes, integrate them in a first release of the MegaM@Rt2 framework and run an initial set of validation scenarios. Based on the obtained results, in the third phase, we will integrate and validate the technical solutions, provide final validation and experience reports from the use cases (as well as a final management report). In parallel, the dissemination (academic or industrial, including the relation with the standardization organizations such as the Object Management Group, OMG) and exploitation (e.g., consortium and individual business plans) activities will be conducted in WP6. The general project management and reporting activities will be performed under the umbrella of WP7.

7. Work Package Descriptions and Deliverables

We will now present a brief description of each WP along with briefly describing some already produced deliverables.

7.1. WP1 - Case Study Requirements Analysis and Architecture Specification

This WP gathers the work on the case studies definition and requirements analysis (by end-users) with the global architecture and road map specification (by technology providers). The industrial partners will set real requirements for research and technology providers. They will closely collaborate and be integrated in the development teams, providing regular feedback on the elaborated technologies. This WP also concentrates on the validation scenarios, i.e. end-to-end demonstrators for the MegaM@Rt2 solutions in varied industrial contexts. End-users will develop methods for gathering the data needed for qualitative and quantitative verification of MegaM@Rt2 achievements. They will run the related experiments in a cost-efficient manner, and will provide representative evaluation of the technologies for large scale usage. From their side, the technology partners will define the architecture and a detailed road map for the technical developments.

7.1.1. Examples of already produced deliverables in WP1

Industry Requirements Specification: This delierable marks the first step of activities for the MegaM@RT2 case study providers. They have defined the case studies to be developed during the project, exposed their current practices, organized the capabilities of the MDE framework they plan to use according to development scenarios, and mapped such scenarios on the time line of the project execution.

On the basis of all above, the case study providers have expressed their end-user requirements for the improved MDE framework that MegaM@RT2 aims to provide. This deliverable is also the first step of the collaboration between the case study providers and the technology providers. This deliverable is also an input to the roadmap development of the MegaM@Rt2 framework, where the capabilities of the baseline tools will be matched to the requirements from the case study providers.

A total of 9 case study providers have given their concrete requirements and their expectations from the MegaM@Rt2 framework (Table 1).

Architecture Specification and Roadmap: This deliverable defines the initial vision of the global architecture of the MegaM@Rt2 framework. As a starting point, we have described the conceptual tools as well as the individual tools by partners. The initial version of the deliverable has concentrated on the following aspects:

- *High-level requirements* to identify the features, goals and objectives of each tool component. These technology requirements will serve as the starting point for the ongoing refinement, elicitation and traceability work that links the tools and methods development with case study development.
- *Functional interfaces* that define the high-level services of the tools as well as the integration points. We took attention to extract common interfaces with the goal to match the tools that may easily collaborate.
- *Subordinates* that are high-level parts of the tool components that help to

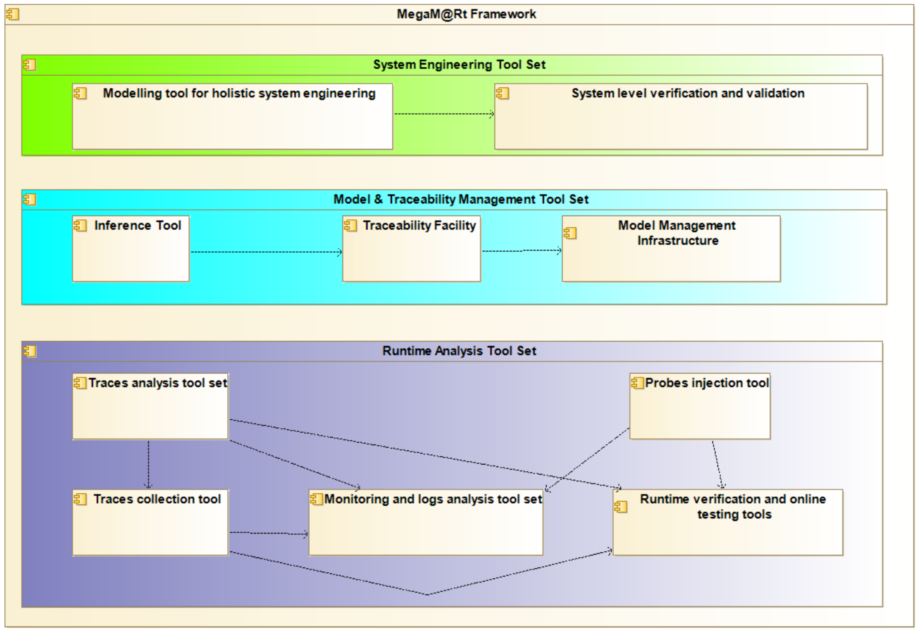

Figure 4: MegaM@Rt2 Framework Architecture Overview

understand better the tool functionality.

- *Deployment* to refer the deployment platforms by paying attention to extract the commonalities that would help to identify the facility for an integrated solution.

We have selected Modelio as the common platform for architecture modelling, primarily because SOFTEAM (MegaM@Rt2 technical leader) is an active contributor to Modelio development and has all the technical and support means to help partners to model in a productive way. Figure 4 shows a high-level architecture of MegaM@Rt2 framework in Modelio. The modeled framework regroups several interconnected tool sets including tool sets for *Holistic System Engineering*, *Model and Traceability Management* as well as for *Runtime Analysis*.

7.2. WP2 - MegaM@Rt2 System Engineering

This WP gathers the activities related to the definition of the required Domain Specific Languages (DSLs) to support model-based system design, and of the methods and tools to develop integrated system models. One of the strongest points of model-based approaches lies in the support for separation of concerns and definition of specific architectural views. Specific views focus on specific areas of the development from system to software level, including the system functional, logical and physical decomposition, identification of software and hardware components, definition of functional and non-functional properties, software architecture, data, behavior and algorithmic modeling. This WP concentrates on all the modelling and tooling aspects of MegaM@Rt2. The goal is first to provide the foundations for WP3 and WP4, and later, to design, develop and support the MegaM@Rt2 system engineering tool set to be used by industrial partners in WP5.

7.2.1. Example of already produced deliverable in WP2

Foundations for Model-driven Design Methods: This deliverable provides the foundations for the design of the MegaM@Rt2 tool chain. Its objective is to analyse the state-of-the-art in terms of both research approaches and existing modelling solutions and tools in the context of model-based continuous development. Within this task, relevant existing DSLs and modelling technologies have been identified and presented, and the possibilities for their utilization, extension and/or integration within MegaM@Rt2 have been analysed. The objective is to provide an overview of the current state of practice, and define the concepts, features and principles that will be the basis for the development of the MegaM@Rt2 design solutions. In particular, the foundation for models, DSLs and their semantics have been addressed. The content of the deliverable has been organized around three main topics: (i) Systems Modelling, (ii) Verification and Validation and (iii) Modelling Methodologies. The first one focuses on standard modelling languages and DSLs, state-of-the-art modelling tools and environments, and methodologies towards the participatory development of DSLs. The second topic covers automatic or semi-automatic solutions for the verification and validation of MDE artefacts (e.g., models, transformations). Finally, the third topic covers different state-of-the-art modelling methodologies. The deliverable ends with a comprehensive catalog of the solutions offered by the different tool providers to all the other members of the consortium. For further information, interested readers can access the full text of the deliverable from the project website[3].

7.3. WP3 - MegaM@Rt2 Runtime Analysis

This WP focuses on the usage and definition of models at runtime level, and on the associated techniques or methods. Models at runtime can be designed or obtained from the system itself. For instance, logging or monitoring the system under the form of models can be performed jointly with the system execution and can help in ensuring a correct system execution. Afterwards, such models can also be analyzed to enhance design models from WP2 and are thus entries of the tools and methods of WP4. Verification and validation issues can be managed directly at runtime, enabling the detection of problems that can be solved at runtime or propagated back to design level. This can be achieved by checking the expected behavior according to functional and non-functional properties embedded in the design models, or by analyzing jointly runtime models with the actual system execution to determine if the system fulfills its specifications. To this intent, this WP will notably provide on-line testing and verification techniques.

7.3.1. Example of already produced deliverable in WP3

Foundations for Model-Based Runtime Methods: This deliverable provides a succinct overview of the foundations of model-based runtime methods and technologies in order to support the innovation tasks of WP3 of the MegaM@Rt2 project.

[3] `http://hdl.handle.net/20.500.12004/1/P/MMART2/D2.1`

The deliverable discusses how runtime artifacts are obtained from design artefacts and from execution logs. In the first category, we overview approaches for generating run time code and models from design models via code generation and, respectively, model transformations. In the second category, we discuss approaches for creating or improving the runtime artifacts by analyzing the runtime execution logs of the system via methods like machine learning and data analytics.

We also discuss how runtime artifacts are used at runtime either by executing them as part of the system at runtime or by using them to generate tests or monitor the system during its operation.

Throughout this deliverable, we have scrutinized the state-of-the art, the state-of-practice, and the baseline technologies which are available for the project participants. To this extent, the deliverable has investigated current methods and tools for their benefits and existing limitations. The results of this deliverable are meant to lay the basis for defining new concepts, methods and tools for coping with these limitations and successfully deploying runtime methods to industrial settings. This deliverable also provides input for the specification of the MegaM@Rt2 runtime tools to support automated code generation and model execution, log analysis, runtime verification and testing activities. The deliverable also includes a collection of relevant solutions and tools provided by the MegaM@Rt2 consortium members as baseline technologies in the project. For further information, interested readers can access the full text of the deliverable from the project website[4].

7.4. WP4 - MegaM@Rt2 Global Model and Traceability Management

This WP focuses on megamodelling, also called global model management, in which models for design time (WP2) and models for runtime time (WP3) are to be managed and aligned all together. This relies on the base notion of a megamodel [22], a model that intends to describe the metadata on the different

[4] `http://hdl.handle.net/20.500.12004/1/P/MMART2/D3.1`

models involved in a given engineering process, as well as the related interrelationships and corresponding artifacts (transformations, generators, etc.). Such a (mega)model can be navigated and queried at any time in order to retrieve or compute the required information, notably as far as traceability between models is concerned.

In the context of MegaM@Rt2, a particular focus is put on various scalability topics: not only the size of the models is larger, but there is also a larger number of model users with different roles; there are various kinds of languages (DSLs) involved for different needs, including e.g. user interface (UI) related languages, and various transformations related to them. The second (and directly related) focus of this WP is on traceability between design time and runtime, as not in all cases the same model can be used for both purposes. WP4 also provides implementation of the tooling for scalable megamodeling/traceability and guidelines for their deployment and practical use in case studies. WP4 is designed to deliver its results incrementally, notably by collecting progressively feedback on the developed features from their application to the project use cases. Figure 5 summarizes the focus of WP4.

7.4.1. Example of already produced deliverable in WP4

Foundations for Model Management and Traceability:

The main goal of WP4 is to elaborate on the required glue between the artifacts produced in WP2 (e.g., design models) and the ones produced in WP3 (e.g., runtime models). As a result, it is expected to provide a so-called global MegaM@Rt2 Model and Traceability Management framework to be a core part of the MegaM@Rt2 overall solution and to be notably deployed on the projects use cases (among possibly others). As the initial step in WP4, this deliverable thus provides an overall state-of-the-art in terms of existing model management and traceability solutions. It presents the main common principles and approaches related to model storage, querying, handling and linking with others models and modeling artifacts, notably via model views [23] and/or so-called megamodels [24]. It also describes the available traceability and interoperabil-

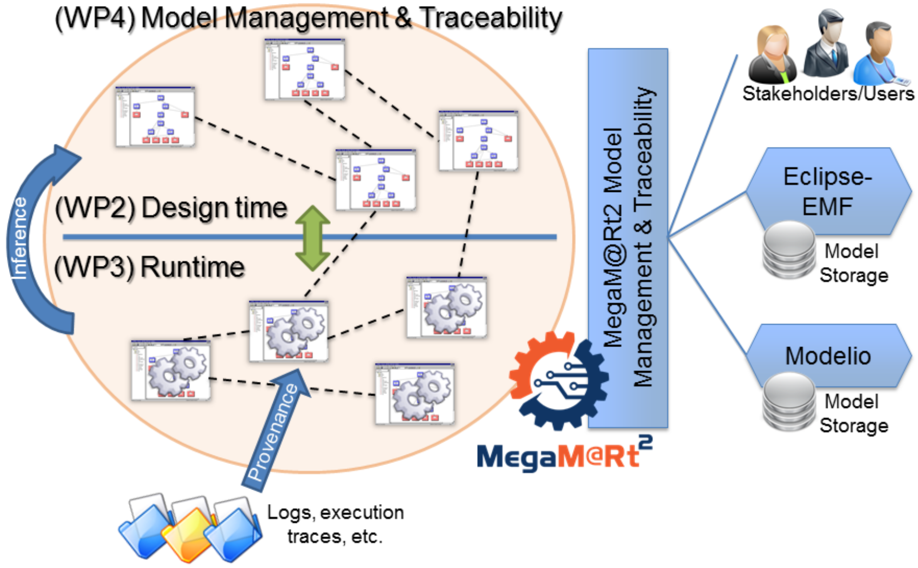

Figure 5: WP4 – Model and Traceability Management approach.

ity solutions [25]. It describes both existing research approaches as well as some more business-oriented tools or environments which are relevant in this given context. Finally, it ends with a list of technical solutions provided by the projects partners. All along the deliverable, a particular importance has been given to aspects related to the scalability of the available solutions.

The main purpose of this deliverable is to prepare the work for specifying the Model and Traceability Management framework to be developed and further used in MegaM@Rt2. Its goal is also to help selecting some of the key problems to be addressed while implementing this framework in the future. Among others, the following big challenges have been identified as important in their respective research areas: scalable model storage and querying, well-synchronized and verified model views, performant and decentralized global model management, efficient integration of inter-model traceability and interoperability support. For further information, interested readers can access the full text of the deliverable from the project website[5].

7.5. WP5 - Integration, Case Study Development and Evaluation

This WP provides specific industrial case studies from different domains such as aeronautics, railway, construction and telecommunication. The main goal of WP5 will be to integrate the different technical developments realized in WP2, WP3 and WP4. It will also be in charge of conducting controlled experiments on the case study partner premises, as defined in WP1. Partners in WP5 will perform a preliminary evaluation as feedback for WP2-3-4, and a strong interaction between technology and use case providers is expected. Finally, WP5 will perform the final integration and consolidation of the MegaM@Rt2 solution, as well as the overall validation the obtained results.

7.6. WP6 - Dissemination and Exploitation

This WP concentrates on the project impact and community building activities. These activities will provide a solid base to identify the key stakeholders for sustainable exploitation, dissemination, communication and standardization.

7.6.1. Examples of already produced deliverables in WP6

Public Website and Social Media Presence: A twitter account for the project has been created. Twitter handle is: @megamart2_ecsel. The project website URL is: `https://megamart2-ecsel.eu/`.

Communication Plan: In the initial version of the communication plan, we have identified a preliminary list of stakeholders who would be especially interested in the project and would thus serve as a specific target for our communication and dissemination plan. For further information, interested readers can access the full text of the deliverable from the project website[6].

[5] `http://hdl.handle.net/20.500.12004/1/P/MMART2/D4.1`
[6] `http://hdl.handle.net/20.500.12004/1/P/MMART2/D6.2`

7.7. WP7 - Management

This work package gathers all the activities related to the management of the MegaM@Rt2 project and its consortium. This mostly includes the mandatory official monitoring and reporting tasks (to the ECSEL Joint Unit and the European Commission). The overall objective is to ensure a smooth running of the project and efficient collaborations between all the involved partners. As fundamental to the success of the project, this WP will notably coordinate the establishment of a proper quality plan to be applied to all MegaM@Rt2 results. It will also deal with the important risk management and Intellectual Property (IP) issues that may appear during the course of the project.

7.7.1. Example of already produced deliverable in WP7

Project Management Guide and Quality Plan: The purpose of this deliverable is to present and describe quality standards and procedures to be applied in the internal management and execution of the project. This document is based on the terms and conditions established in the Grant Agreement signed by the ECSEL-JU. This deliverable describes the management roles and functions, the decision and control procedures, the processes and resources for ensuring the quality of project deliverables.

This deliverable is intended to be used by the project management team and the work package leaders, as well as people who are directly responsible for producing the deliverables, to ensure the quality assurance of project processes and outputs and to avoid eventual deviations from the project work plan.

8. Conclusion

This paper presented the MegaM@Rt2 ECSEL project. It notably provided the global context and motivation for this project, introduced its mission and targeted objectives, described its general organization in terms of work packages and detailed the composition of its large supporting consortium. As explained in this paper, MegaM@Rt2 mainly intends to create a scalable model-based framework for dealing with the continuous development and validation of the software parts of large and complex industrial CPSs. This framework will notably focus on relating together the actual executions of these systems (i.e., runtime) with the way they are currently specified, developed and maintained (i.e., design time). While there is already quite a lot of support for these two dimensions separately, there is currently no real support for an efficient integration and feedback loop between design time and runtime. We plan to practically realize this by providing the required management and traceability support between all the involved models (both at design time and runtime). The obtained results will be experimented on 9 different use cases covering different industrial domains such as aerospace, railway, telecommunication, networks and construction equipments. In addition to scientific progress in the CPSs and modeling/MDE domains, industrial partners are expected to gain concrete benefits in terms of improvements to their system reliability and decrease in development and maintenance costs.

Acknowledgment

This project 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 European Union's Horizon 2020 research and innovation program and from Sweden, France, Spain, Italy, Finland & Czech Republic.

References

[1] P. Wallin, S. Johnsson, J. Axelsson, Issues Related to Development of E/E Product Line Architectures in Heavy Vehicles, in: 42nd Hawaii International Conference on System Sciences (HICSS), 2009.

[2] S. K. Khaitan, J. D. McCalley, Design Techniques and Applications of Cyberphysical Systems: A Survey, IEEE Systems Journal 9 (2) (2015) 350–365.

[3] A. G. Kleppe, J. Warmer, W. Bast, MDA Explained: The Model Driven Architecture: Practice and Promise, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2003.

[4] B. Anda, K. Hansen, I. Gullesen, H. K. Thorsen, Experiences from Introducing UML-based Development in a Large Safety-critical Project, Empirical Softw. Engg. 11 (4) (2006) 555–581.

[5] P. Baker, S. Loh, F. Weil, Model-Driven Engineering in a Large Industrial Context — Motorola Case Study, in: Proceedings of the 8th International Conference on Model Driven Engineering Languages and Systems, Springer-Verlag, 2005.

[6] F. Tomassetti, M. Torchiano, A. Tiso, F. Ricca, G. Reggio, Maturity of Software Modelling and Model Driven Engineering: A Survey in the Italian Industry, in: 16th International Conference on Evaluation Assessment in Software Engineering (EASE 2012), 2012.

[7] M. Petre, "No Shit" or "Oh, Shit!": Responses to Observations on the Use of UML in Professional Practice, Softw. Syst. Model. 13 (4) (2014) 1225–1235.

[8] T. Gorschek, E. Tempero, L. Angelis, On the Use of Software Design Models in Software Development Practice: An Empirical Investigation, J. Syst. Softw. 95 (2014) 176–193.

[9] D. Di Ruscio, R. F. Paige, A. Pierantonio, Guest Editorial to the Special Issue on Success Stories in Model Driven Engineering, Science of Computer Programming 89 (PB) (2014) 69–70.

[10] L. Baresi, C. Ghezzi, The Disappearing Boundary Between Development-time and Run-time, in: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research (FoSER), ACM, 2010.

[11] B. Fitzgerald, K.-J. Stol, Continuous Software Engineering: A Roadmap and Agenda, Journal of Systems and Software 123 (2017) 176 – 189.

[12] D. Brahneborg, W. Afzal, A. Causevic, D. Sundmark, M. Björkman, Round-Trip Time Anomaly Detection, in: Proceedings of the 2018 ACM/SPEC International Conference on Performance Engineering, ACM, 2018.

[13] D. Brahneborg, W. Afzal, A. Causevic, A Black-Box Approach to Latency and Throughput Analysis, in: 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), 2017.

[14] W. Afzal, H. Bruneliere, D. D. Ruscio, A. Sadovykh, S. Mazzini, E. Cariou, D. Truscan, J. Cabot, D. Field, 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, in: Proceedings of the 2017 Euromicro Conference on Digital System Design (DSD'17), 2017.

[15] W. Afzal, R. Torkar, R. Feldt, A Systematic Review of Search-based Testing for Non-functional System Properties, Information and Software Technology 51 (6) (2009) 957 – 976.

[16] W. Afzal, R. Torkar, R. Feldt, A Systematic Mapping Study on Non-functional Search-based Software Testing, in: Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering (SEKE'08), 2008.

[17] B. S. Ahmed, K. Z. Zamli, W. Afzal, M. Bures, Constrained Interaction Testing: A Systematic Literature Study, IEEE Access 5 (2017) 25706–25730.

[18] V. Muttillo, G. Valente, F. Federici, L. Pomante, M. Faccio, C. Tieri, S. Ferri, A Design Methodology for Soft-core Platforms on FPGA with SMP Linux, OpenMP Support, and Distributed Hardware Profiling System, EURASIP Journal on Embedded Systems 2016 (1) (2016) 15.

[19] G. Valente, V. Muttillo, L. Pomante, F. Federici, M. Faccio, A. Moro, S. Ferri, C. Tieri, A Flexible Profiling Sub-System for Reconfigurable Logic Architectures, in: 2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP), 2016.

[20] A. Moro, F. Federici, G. Valente, L. Pomante, M. Faccio, V. Muttillo, Hardware Performance Sniffers for Embedded Systems Profiling, in: 2015 12th International Workshop on Intelligent Solutions in Embedded Systems (WISES), 2015.

[21] S. E. Cross, T. Felis, A Systems Engineering Approach to Systematic Innovation in an Industry-University Collaboration, in: 2016 IEEE European Technology and Engineering Management Summit (E-TEMS), 2016.

[22] J.-M. Favre, T. NGuyen, Towards a Megamodel to Model Software Evolution Through Transformations, Electronic Notes in Theoretical Computer Science 127 (3) (2005) 59–74.

[23] H. Bruneliere, E. Burger, J. Cabot, M. Wimmer, A Feature-based Survey of Model View Approaches, Software & Systems Modeling (2017) 1–22.

[24] R. Hebig, A. Seibel, H. Giese, On the Unification of Megamodels, Electronic Communications of the EASST 42.

[25] N. Aizenbud-Reshef, B. T. Nolan, J. Rubin, Y. Shaham-Gafni, Model Traceability, IBM Systems Journal 45 (3) (2006) 515–526.