The Functional Mockup Interface - seen from an industrial perspective

Christian Bertsch Elmar Ahle Ulrich Schulmeister
Robert Bosch GmbH, Corporate Sector Research and Advance Engineering,
Robert-Bosch-Strasse 2, 71701 Schwieberdingen, Germany
christian.bertsch@de.bosch.com elmar.ahle@de.bosch.com ulrich.schulmeister@de.bosch.com

Abstract

The demand for model exchange between development partners will grow during the next years. The Functional Mockup Interface (FMI) is a well received tool independent approach for model exchange. The Original Equipment Manufacturers (OEM) have committed themselves to support FMI as exchange format for simulation models. Therefore, the FMI is a promising candidate to become the industry standard for model exchange and cross-company collaboration. In this paper, the FMI standard is evaluated from an industrial perspective.

Keywords: FMI, industrial perspective, problem classification

1 Introduction

There is a strong trend of virtualization in engineering, where simulation replaces real testing in order to develop faster, cheaper and with more system understanding. Therefore, system simulation is used in different phases of product development und gains importance within the entire product development cycle. The design of future products is done on the basis of simulation models.

1.1 Modeling and Simulation at Bosch

However, the simulation landscape in the field of system and component design in industry is very heterogeneous. At Bosch, there are more than 100 simulation tools with incompatible model representations in use, some of them are preferred tools for different physical or engineering domains. The picture becomes even more complex when looking at external partners such as Original Equipment Manufacturers (OEM) with a different set of preferred simulation tools. Today, there exist only proprietary exchange formats that are limited in functionality and only applicable to a limited number of tool combinations.

1.2 Approaches for modeling of complex systems and model exchange

There are two complementary approaches for modeling complex systems:

White box modeling: Modeling the entire system with one modeling language. This requires a modeling language that is suitable for different physical domains. This approach offers the possibility for deep system understanding by equation-based, object-oriented modeling and symbolic manipulation. MODELICA is an example for that approach.

Black box model exchange: Defining an interface for model exchange for standardized, tool-independent exchange format for simulation models is a complementary approach. It offers a way to cope with a heterogeneous simulation tool environment and allows using specialized tools for different physical domains. This approach offers also the possibility for know-how protection for model exchange in distributed collaborative system engineering. The Functional Mockup Interface (FMI) is a promising candidate to become the industry standard for this kind of model exchange and cross-company collaboration.

In the following, the black box model exchange with FMI is evaluated from an industrial perspective. Requirements on such an interface for industrial applications are

- *standardization* of the model interface (in order to be tool-independent),
- *availability* of a significant number of supporting tools,
- *easy-of-use* of the interface for simulation engineers,
- *adoption* of the standard in the specific industry domain,

- *accompanying documentation* and a reference process for the exchange of such black box models, and
- *maturity* of such an interface, i.e.,
  - no errors when importing FMUs from other sources and
  - reliable simulation results when using such an interface.

1.3 The Functional Mockup Interface for black box model exchange

The Functional Mockup Interface [1-4] has been developed to meet the requirements listed above. The requirement *standardization* and tool-independence is fulfilled by design. There is growing support for FMI by simulation tools (i.e., more than 40 listed on the FMI website), so the *availability* is given (although some often used simulation tools are still missing). The *easy-of-use* to some extent is also by design (e.g., XML-file for model information) and is realized quite well by the majority of exporting and importing simulation tools. The Global Automotive Advisory Group (GAAG) has committed itself to support FMI as exchange format for simulation models [5]. Thus, the broad *adoption* is at least planned. The *accompanying documentation* is addressed in the ProSTEP Smart Systems Engineering Project [6]. In the following the paper will focus on the *maturity* of the FMI standard or the maturity of the implementation respectively.

2 The Maturity of FMI

2.1 Maturity of FMI after the end of the MODELISAR project

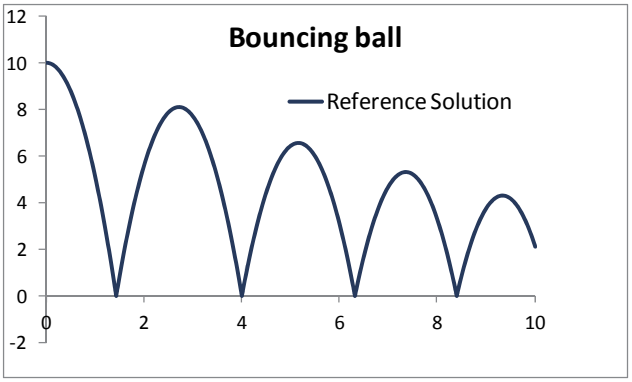
As a result of the MODELISAR project, the FMI standard 1.0 was published and it was implemented by a growing number of modeling and simulation tools. In the MODELISAR project, requirements for FMI were derived from the beginning and tested for industrial applications. The performance of the FMI approach was demonstrated in 24 industrial applications [4]. Additional successful applications have been reported at the 8th and 9th MODELICA Conference, see e.g. [7-10]. However, these examples were realized in some fixed combinations of FMU-exporting and importing tools.

In 2012, an internal benchmark at Bosch with three exporting tools (12 test FMUs exported) and five importing tools showed quite different results. The test examples range from a "model" containing a sine generator only, a bouncing ball, a spring-damper system, an RC circuit to a thermal network. While some combinations worked quite well, other combinations did not work at all. Typical problems at that time were

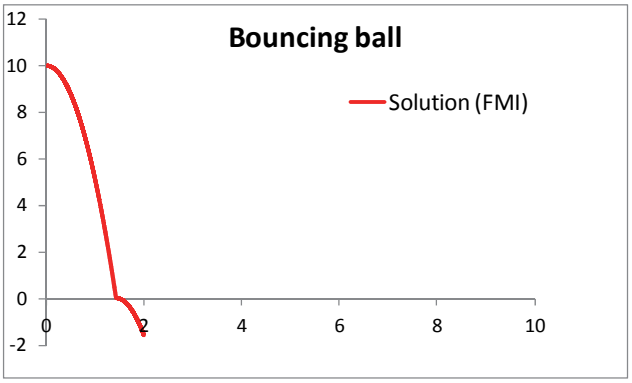
- formal errors in the XML-file,
- errors during initialization,
- memory leakage, and
- different simulation results in different importing tools.

Test results for the bouncing ball example are as depicted in Figure 1. The model was created in tool 1. Then, it was exported as a Functional Mockup Unit (FMU) for model exchange. The import of the FMU in the same simulation tool was possible, but the simulation results were erroneous, see Figure 1. The simulation of the FMU was also not possible in four other simulation tools due to different errors, e.g.,

- failed assertions,
- two importing tools crashed during simulation, or
- unspecified error at import.



(a) Reference solution



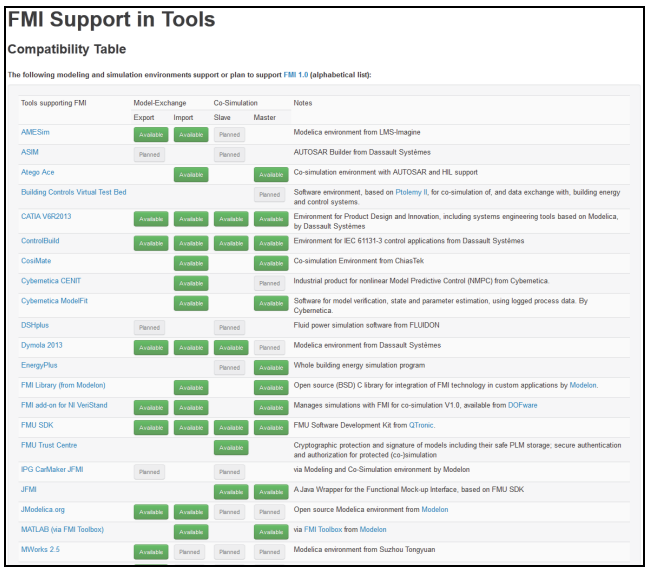
(b) FMU simulation result in tool 1

Figure 1: Comparison of simulation results for a bouncing ball example (status March 2012)

At the same time, every tool could be listed on the FMI website in "green" and claim to support FMI as

DOI 10.3384/ECP1409627

depicted in Figure 2. The number of these tools increased very rapidly, but did not reflect the experience of the internal benchmark.



FMI Support in Tools

Compatibility Table

The following modeling and simulation environments support or plan to support FMI 1.0 (alphabetical list):

Tools supporting FMI	Model-Exchange		Co-Simulation		Notes
	Export	Import	Slave	Master	
AMESim	Available	Available	Planned		Modelica environment from LMS-Imagine
ASIM	Planned		Planned		AUTOSAR Builder from Dassault Systèmes
Atego Ace		Available		Available	Co-simulation environment with AUTOSAR and HIL support
Building Controls Virtual Test Bed				Planned	Software environment, based on Ptolemy II, for co-simulation of, and data exchange with, building energy and control systems.
CATIA V6R2013	Available	Available	Available	Available	Environment for Product Design and Innovation, including systems engineering tools based on Modelica, by Dassault Systèmes
ControlBuild	Available	Available	Available	Available	Environment for IEC 61131-3 control applications from Dassault Systèmes
CosiMate		Available		Available	Co-simulation Environment from ChiasTek
Cybernetica CENIT		Available		Planned	Industrial product for nonlinear Model Predictive Control (NMPC) from Cybernetica.
Cybernetica ModelFit		Available		Available	Software for model verification, state and parameter estimation, using logged process data. By Cybernetica.
DSHplus	Planned		Planned		Fluid power simulation software from FLUIDON
Dymola 2013	Available	Available	Available	Planned	Modelica environment from Dassault Systèmes
EnergyPlus			Planned	Available	Whole building energy simulation program
FMI Library (from Modelon)		Available		Available	Open source (BSD) C library for integration of FMI technology in custom applications by Modelon.
FMI add-on for NI VeriStand	Available	Available		Available	Manages simulations with FMI for co-simulation V1.0, available from DOFware
FMU SDK	Available	Available	Available	Available	FMU Software Development Kit from QTronic.
FMU Trust Centre			Available		Cryptographic protection and signature of models including their safe PLM storage; secure authentication and authorization for protected (co-)simulation
IPG CarMaker JFMI	Planned		Planned		via Modeling and Co-Simulation environment by Modelon
JFMI			Available	Available	A Java Wrapper for the Functional Mock-up Interface, based on FMU SDK
JModelica.org	Available	Available	Planned	Planned	Open source Modelica environment from Modelon
MATLAB (via FMI Toolbox)		Available		Available	via FMI Toolbox from Modelon
MWorks 2.5	Available	Planned	Planned	Planned	Modelica environment from Suzhou Tongyuan

Figure 2: Beginning of list of tools supporting FMI (source: https://www.fmi-standard.org, March 2012)

2.2 A classification of possible problems in FMI-based simulation

The challenges encountered in the benchmark are classified and ordered with increasing maturity of the FMI standard:

I. Inconsistencies of the standard
- Source of problem: Standard
- Examples: Ambiguities regarding naming conventions and structure of ZIP-file, XML model-description file

II. Formal errors in the FMU-File
- Source of problem: Exporting tool
- Examples: Errors in XML and internal directory structure of ZIP-file

III. Formal interaction of importing tool with FMU
- Source of problem: Exporting or importing tool
- Examples: Errors in calling functions; memory leakage

IV. Simulation of one FMU (with solver of importing tool in case of model exchange)
- Source of problem: Exporting or importing tool
- Examples: Incorrect initialization; incorrect simulation result; incorrect event handling

V. Simulation of multiple connected FMUs with solver for model exchange
- Source of problem: Solver or master algorithm of exporting/importing tool, overall system divided into several FMUs at unsuitable location
- Examples: Incorrect treatment of algebraic loops; numerical problems with FMUs with different time constants (stiffness)

These problem categories are summarized and ordered in Figure 3. The problem classes correspond to different stages in the simulation process. If an error occurs in one problem class, typically the simulation fails or the results are not reliable.

At the beginning of 2012, the FMI standard 1.0 was quite mature so that problem class I was not a big issue. But there occurred several problems corresponding to problem classes II to IV for exchanging even single FMUs between different tools and simulating them. Only the problem classes I to IV are FMI-specific; problem class V corresponds to typical simulation challenges of modular simulation with causal interfaces or co-simulation. Some of the problems in FMI-based simulation might be observed for the first time, because models have neither be exchanged nor been coupled in the past.

3 Implemented measures to improve maturity

The maturity issues of FMI-based simulation were addressed to the FMI community at the MODELICA/FMI meetings beginning early 2012. The discussions and input from other companies and users resulted in the call for quotation of an FMU

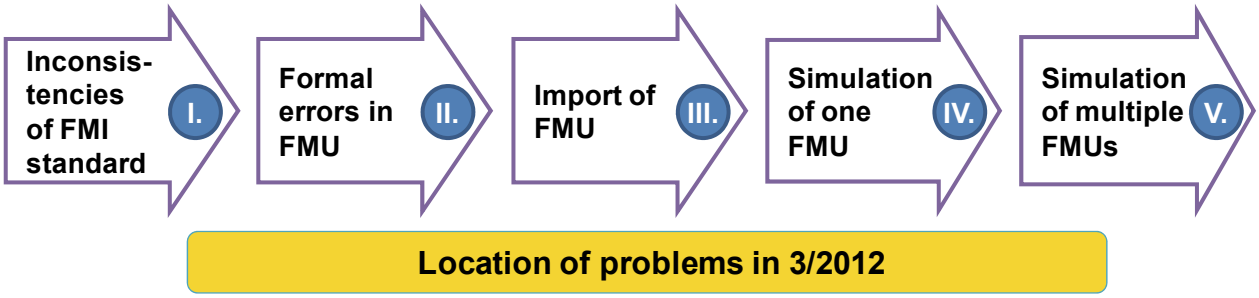


Figure 3: Classification of problems in FMI-based simulation

Compliance Checker and later of FMI Cross Checking rules. Bosch's application for membership in the FMI Steering Committee was accepted in January 2013. Since then Bosch is actively contributing at FMI design and steering committee meetings.

3.1 FMU Compliance Checker

The FMU Compliance Checker [11] is a software tool that was initiated by the MODELICA Association by a call in 04/2012 and was implemented by the company MODELON AB. It addresses problem classes II, III, and also partly IV (the tool contains only a very simple solver). In the first release only one FMU could be called and no inputs to the FMU could be provided. With a later release inputs to FMUs can be provided as CSV-files. The FMU Compliance Checker is a very valuable tool to check for formal errors of the FMU (see Figure 4).

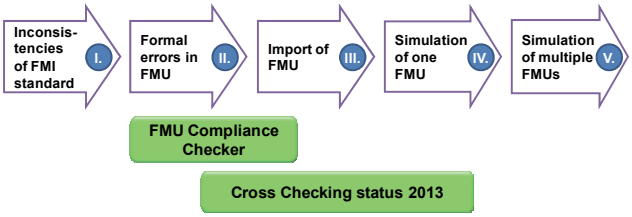


Figure 4: Focus of FMU Compliance Checker and FMI Cross Checking

3.2 FMI Cross Checking

The FMI Cross Check Rules [12] were approved in February 2013 and focus on testing the quality of implementations of exporting and importing tools. It is defined how many FMUs an exporting tool must publish and for how many of these FMUs an importing tool must publish simulation results in order to get listed as "available" on the FMI-tools website [3]. As at the moment only single FMUs are tested, FMI Cross Checking at the moment focuses on problem class III and IV as depicted in Figure 4. As a side effect it will also show problems of class I and II.

Figure 5 shows a screenshot of the FMI Tools website [3] and an entry in green refers to an exporting or importing tool that has successfully passed FMI Cross Checking. An entry in orange shows that the corresponding FMI export or import is claimed to be available but is not tested according to the FMI Cross Check Rules.

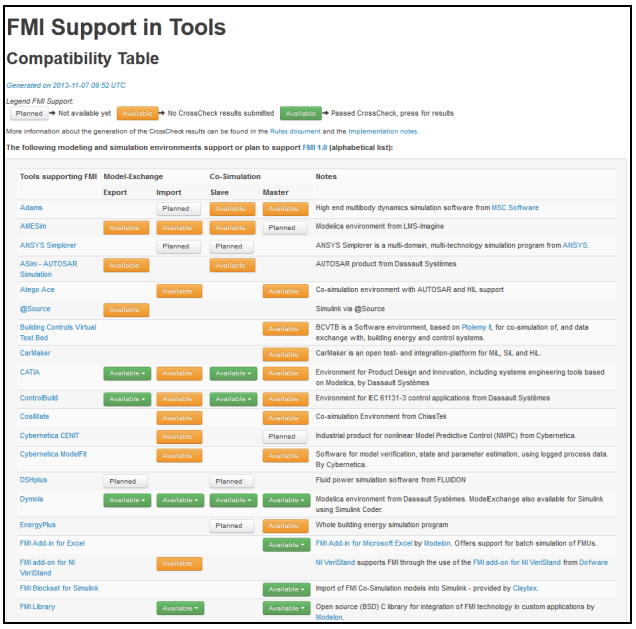


Figure 5: Beginning of list of tools supporting FMI (source https://www.fmi-standard.org, status 11/08/2013)

At first glance, it is disappointing that there are so many "orange" entries of exporting tools that have not yet provided test FMUs and of importing tools that have not yet published simulation results. On the other hand this makes it transparent that there are combinations of FMU-exporting and importing tools that are tested and other that are not. This is exactly what was intended with the FMI Cross Checking rules and helps the user to select tools that he can rely on.

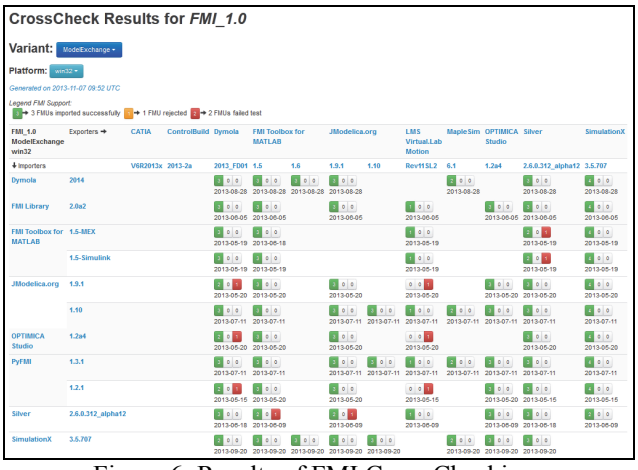


Figure 6: Results of FMI Cross Checking, (source https://www.fmi-standard.org, status 11/8/2013)

For the tools that participate in FMI Cross Checking, detailed results are published on the FMI website, as depicted in Figure 6. Thus, FMI Cross Checking is very valuable for industrial users, as they are supported with information, which combinations of exporting and importing tools have already

DOI 10.3384/ECP1409627

been tested successfully for some applications. FMI Cross Checking is a source to detect improvement potential of the simulation tools.

4 Proposed next steps

The actions that we have seen until now address mainly the problem classes II to IV. In order to address "real-life problems", FMI Cross Checking should be extended to more tools and for multiple FMUs. The (co-)simulation techniques for importing tools should be improved and the improvements of FMI 2.0 should be implemented soon.

4.1 Extension of FMI Cross Check Rules

There should be more exported test FMUs provided by much more exporting tools und much more results of importing tools be published. FMI Cross Checking should be extended to *connected FMUs*. These FMUs could come from different exporting tools.

A simple test case is shown in Figure 7: An electrical motor with a gear and load torque (plant model) with PI controller (model of the software). The plant model and the control model are created in different MODELICA based simulation tools, then (separately) exported as FMUs and re-imported in different tools. This is a simple test example for a closed loop control, where the control software FMU might come from a different source than the plant model FMU. In some cases this worked already well from the beginning (2012), in other cases bugs have been fixed.

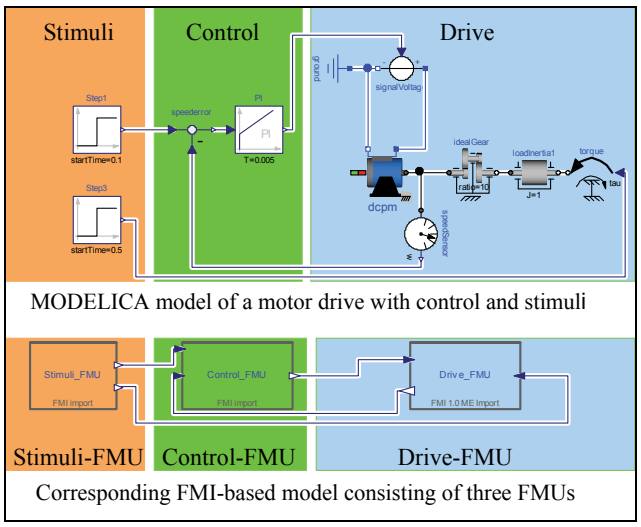


Figure 7: Example of connected FMUs exported from a MODELICA model

Further test examples of connected FMUs should be provided, e.g.,

- a stiff mechanical system consisting of various spring and damper elements exported as several FMUs, or
- a hybrid powertrain containing of different FMUs:
  - Simple map-based model of a combustion engine,
  - mechanical powertrain with gears and clutches,
  - electrical machine,
  - power electronics, and
  - control software model for the electrical machine.

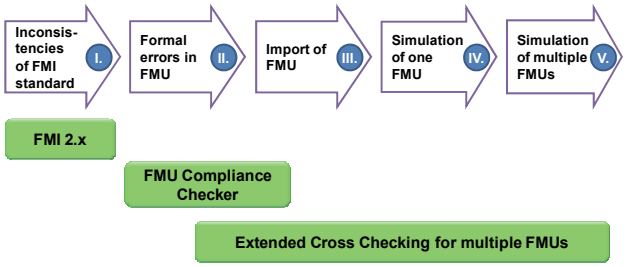


Figure 8: Focus of FMI 2.x, FMU Compliance Checker and extended FMI Cross Checking

4.2 Extension of test cases to multiple connected FMUs

In order to test multiple connected FMUs in different tools easily a standardized file format for the definition of the set-up is proposed. This file should contain the following information:

- Simulation settings (e.g., simulation duration),
- list of the included FMUs,
- parameters of the included FMUs, and
- list of connections (i.e., which output of an FMU is connected to which input of another FMU).

Such a file should be preferably implemented as an XML file. All stimuli can also be defined in FMUs or provided as CSV file, as it is already possible for single FMUs within cross-checking. All files for a particular test should be either provided in a ZIP-file or a directory. With the help of such a standardized file describing the connections of multiple FMUs, the testing of connected FMUs can be done automatically in batch runs.

4.3 Improvement of (co-)simulation techniques of importing tools

New co-simulation techniques and master algorithms are proposed in the scientific community, see e.g. [13-15]. These new methods will improve co-simulation techniques. Also for model exchange new solver techniques should be developed that can handle multi-rate systems divided in several FMUs. The (co-)simulation capabilities of the importing tools for multiple FMUs can be compared by extended FMI Cross Checking.

4.4 Improvements of the FMI standard with FMI 2.0

Problem class I is addressed by bug fixes of the FMI standard. Maturity issues of the FMI standard 1.0 are not considered problematic, but additionally to new features, there will be some minor inconsistencies eliminated. FMI Cross Checking is an integral part of the introduction phase of FMI standard 2.0. Hence, it is expected that possible problems in the standard (problem class I) and the implementations (problem classes II to IV) will be detected at an early stage. FMI standard 2.0 will be only released, when a significant number of test implementations are available.

5 Outlook

5.1 Usage of FMI at Bosch

FMI enables the exchange of models between tools where this either had not been possible in the past or only by proprietary exchange formats (resulting in some cases in additional license fees). Additionally, FMI could replace existing in-house solutions of tool couplings and co-simulation. The use of a widely accepted standard is much more effective than developing and maintaining special interfaces.

5.2 FMI-based collaboration with OEMs

One motivation of support FMI is the increasing demand to exchange simulation models between OEM and suppliers [10]. While FMI as a technical standard is right on track, there are other points to be addressed such as model exchange process and accompanying documentation. This is developed in the ProSTEP Smart Systems Engineering Project (SSE) [6] in cooperation with OEMs. The main question there is how collaborative simulation-based engineering based on FMI can work. FMI is the best available approach for tool-independent model exchange and co-simulation. FMU Compliance Checking and FMI Cross Checking address technical problems that existed in the past. FMI Cross checking should be extended to more complicated examples and also with multiple FMUs. Then, FMI can become the technical basis for model-based collaborative engineering in a heterogeneous tool environment with different partners.

References

[1] Blochwitz, T. et al., The Functional Mockup Interface for Tool independent Exchange of Simulation Models, In: Proceedings of the 8th International Modelica Conference 2011, Dresden, Germany

[2] Blochwitz, T. et al., The Functional Mockup Interface 2.0: The Standard for Tool independent Exchange of Simulation Models, In: Proceedings of the 9th MODELICA Conference 2012, Munich, Germany

[3] FMI project website, https://www.fmi-standard.org/

[4] Chombard, P., Multidisciplinary modeling and simulation speeds development of automotive systems and software, ITEA2 innovation report, published online: http://www.itea2.org/project/index/view/?project=217

[5] Chombard, P., Standard FMI pour l'échange et la co-simulation de modèles multi-disciplinaires, LMCS Conference 2012, http://www.acsysteme.com/fr/lmcs-2012

[6] ProSTEP Smart Systems Engineering (SSE) Project, http://www.prostep.org/

[7] Abel, A. et.al., Functional Mockup Interface in Mechatronic Gearshift Simulation for Commercial Vehicles, In: Proceedings of the 9th MODELICA Conference 2012, Munich, Germany

[8] Sun Y., Vogel S., Steuer H., Combining Advantages of Specialized Simulation Tools and Modelica Models using Functional Mockup Interface (FMI), In: Proceedings of the 8th MODELICA Conference 2011, Dresden, Germany

[9] Chrisofakis E., Junghanns A., Kehrer C., Rink A., Simulation-based development of automotive control software with Modelica, In: Proceedings of the 8th MODELICA Conference 2011, Dresden, Germany

DOI 10.3384/ECP1409627

[10] Schneider, Stefan-Alexander, Virtual System Prototyping in Automotive Industry and the role of FMI, 7th MODPROD Workshop on Model-Based Product Development 2013, Linköping, Sweden

[11] FMU Compliance Checker, https://www.fmi-standard.org/downloads

[12] FMI Cross Check Rules, https://svn.fmi-standard.org/fmi/branches/public/Cross Check_Results/

[13] Schierz, T., Arnold, M., Clauß C., Co-simulation with communication step size control in an FMI compatible master algorithm, In: Proceedings of the 9th MODELICA Conference 2012, Munich, Germany

[14] Petridis, K., Klein, A., Beitelschmidt, M.: Asynchronous method for the coupled simulation of mechatronic systems. In: PAMM Volume 8 (2008), Nr. 1

[15] Petridis, K.: Synchrone und asynchrone Verfahren zur gekoppelten Simulation mechatronischer Systeme, TU Dresden, Dissertation, to appear