

Release Presentation

ASAM AE XIL-MA 2.0.2

Generic Simulator Interface for Simulation Model Access

2016 / 02 / 22

Agenda

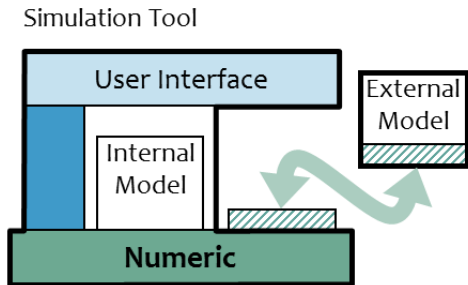
- ▶ **Motivation and Background of XIL-MA**
- ▶ **Introduction and General Concepts**
- ▶ **What's New?**
- ▶ **Deliverables**
- ▶ **Compatibility**

Motivation for XIL-MA

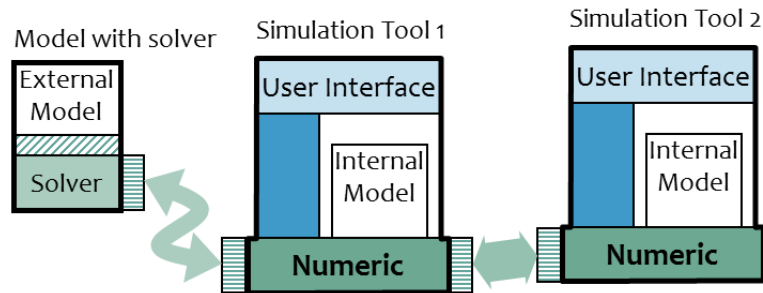
Cooperation of ASAM XIL and ITEA 2 Project MODELISAR

- ▶ European Project MODELISAR (2008 – 2011) was setup to develop a set of open interface standards for simulators

1. FMI for Model Exchange

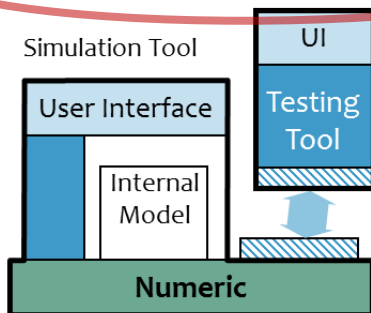


2. FMI for Co-Simulation



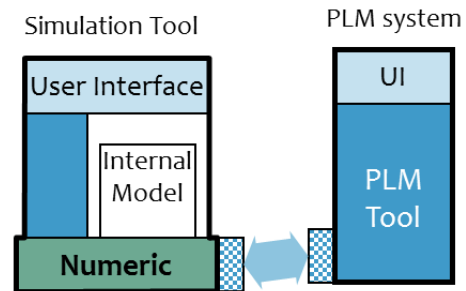
XIL-MA

3. FMI for Applications, e.g. Testing



Done as a joint initiative between MODELISAR and ASAM XIL

4. FMI for PLM



Background on FMI (1)

Result of MODELISAR: Functional Mockup Interface FMI

▶ Open standard

- Free downloads on FMI web page:
<https://www.fmi-standard.org/downloads>

Downloads

FMI - Version 2.0

FMI for Model Exchange and Co-Simulation

This is the second version of the Functional Mockup Interface standard (FMI). It is a major enhancement compared to FMI 1.0, where the FMI 1.0 Model Exchange and Co-Simulation standards have been merged, and many improvements have been incorporated, often due to practical experience when using the FMI 1.0 standards. New features include: Parameters can be changed during simulation, the complete FMU state can be saved, restored and serialized, directional derivatives with respect to states and inputs can be computed, the structure of the partial derivatives with respect to states and inputs can be given (to support large systems), algebraic loops over FMUs are now supported in all modes (initialization, event, continuous-time) for Model Exchange, allowing for example improved initialization.

Version 2.0 was released on July 25, 2014.

[Download Complete Package](#)[Download Specification Only](#)

ASAM XIL-MA

At the beginning of the ITEA2 project MODELISAR (the project in which FMI was initially developed) it was planned to develop FMI for Applications. The intention was to specify an API for simulation tools that allows the access of model parameters, stimulations and simulation results and to control simulation experiments as well. The API was supposed to provide tool independent access to simulation computations for optimization tools, test management and workflow definition tools.

The project group noticed the parallel work of the ASAM XIL standardization group. Thus, both groups came to the conclusion not to develop a parallel standard but to cooperate. The result of this cooperation is available as ASAM XIL-MA, which is a subset from ASAM XIL standard. It contains the model access port specification as well as necessary common functionality for a proper operation of the model access. XIL-MA is open to public and the documentation of the standard can be obtained from ASAM for free without membership.

<https://www.fmi-standard.org/related>

Background on FMI (2)

Functional Mockup Interface standard is going its way

- ▶ Broadly supported by simulation tools, see FMI web page: <https://www.fmi-standard.org/tools>



- ▶ Continuous development on future releases by a group as a Modelica Association Project

- Members include Dassault Systems, Siemens, dSPACE, ETAS, Bosch, Daimler

- ▶ ProSTEP project to utilize FMI as standard exchange format between OEMs and suppliers



- Members include BMW, Bosch, Conti, Daimler, Ford, MAN, several tool vendors

- ▶ Global Automotive Advisory Group for PLM support the ProSTEP initiative

- OEM Members include BMW, Chrysler, Daimler, FIAT, Ford, GM, Nissan, Renault, VW,

FMI Support in Tools
Compatibility Table

Generated on 2015-10-22 20:10 UTC
 Legend: FMI Support: Planned, Available, Not CrossCheck results submitted, Available, Planned, CrossCheck, press for results
 More information about the generation of the CrossCheck results can be found in the Rules document and the Implementation rules.
 The following modeling and simulation environments support or plan to support FMI 2.0 (operational):

Tool	Model-Exchange Support	Co-Simulation Support	Notes
Adams	Planned	Available	High and multi-body dynamics simulation software from MSC Software
ANSYS	Available	Available	ANSYS is a multi-domain, multi-technology simulation program from ANSYS
ANSYS Smpower	Planned	Planned	ANSYS Smpower is a multi-domain, multi-technology simulation program from ANSYS
Asim - AUTOSAR Simulation	Available	Available	AUTOSAR product from Dassault Systèmes
Aspen Hysys	Available	Available	Simulation environment with AUTOSAR and ML support
Autodesk	Available	Available	Software by Autodesk
Building Controls Visual Test Bed	Available	Available	BCVTB is a software environment, based on PlantUML, for co-simulation of, and data exchange with, building energy and control systems
CarSim	Available	Available	CarSim is an open tool and integration platform for MIL, SIL and HIL
CATIA	Available	Available	Environment for Product Design and Innovation, including systems engineering tool based on Modelica, by Dassault Systèmes
ControlDesk	Available	Available	Environment for IEC 61131-3 control applications from Dassault Systèmes
ControlExpert	Available	Available	Co-simulation Environment from Chartwell
Cybernetica	Available	Available	Industry product for nonlinear flexible Control (NAPFC) from Cybernetica
Cybernetica Modelica	Available	Available	Software for model verification, reuse and operational execution, using Modelica, from Cybernetica
dSPACE	Planned	Planned	Full power simulation software from dSPACE
Dynas	Available	Available	Modelica environment from Dassault Systèmes, Modelica/Modelica is available for Simulink using Simulink Coder
EnergPlus	Available	Available	Whole building energy simulation program
FMI Adapter for Simscape	Available	Available	FMI Adapter for Simscape, Simscape for Co-Simulation, Other support for each simulation of FMI
FMI Adapter for MATLAB	Available	Available	FMI Adapter for MATLAB, through the use of the FMI Adapter for Simscape
FMI Support for Simulink	Available	Available	Support of FMI Co-Simulation modes into Simulink - provided by Daimler
FMI Target for Simulink Coder	Available	Available	Code source (SBC) library for integration of FMI technology in custom applications by MathWorks
FMI Target for MATLAB	Available	Available	Support of external FMI for Co-Simulation from Simulink using Simulink Coder - provided by ITT
FMI Target for MATLAB	Available	Available	FMI Target for MATLAB from Chartwell
FMI SDK	Available	Available	FMI Toolkits for MATLAB from MathWorks can be used for MATLAB and Simulink
FMI SDK	Available	Available	FMI Software Development Kit from Chartwell
Hopsan	Available	Available	Hopsan is a free simulation tool developed at Linköping University, with extensive and detailed user support for performing multi-scale processes
IEC61131-3 Co-Simulation	Available	Available	IEC61131-3 is a communication tool developed by Visual Informatics
IEC61131-3	Available	Available	A Java Compiler for the Functional Mockup Interface, based on FMI SDK
Modelica.org	Available	Available	Open source Modelica environment from MathWorks
NI LabVIEW	Available	Available	Visual LabVIEW is a high and multi-body software from NI
OpenSim	Available	Available	Modelica-based modeling and simulation tool from OpenSim
OpenModelica	Available	Available	Modelica environment from Sunojo Technology
PLM LAD/IDE	Planned	Planned	Optical programming environment for measurement, test, and control systems from National Instruments
OpenModelica	Available	Available	Open source Modelica environment from OpenModelica
OPTIMICA Suite	Available	Planned	Modelica environment from MathWorks
PlantUML	Planned	Planned	Software environment for design and analysis of heterogeneous systems
PLM LAD/IDE	Available	Available	For Function via the open source package FMIKit from MathWorks. Also available as part of the Toolkits for Chartwell from Daimler
RealRun	Planned	Planned	High And Multi Flexible Body Dynamics Software from Daimler
Reference Model	Planned	Planned	Reference FMI tool, supported by manufacturers and volunteers to show how to use FMI with their own tools
SCADE Data	Planned	Planned	SCADE Data: hardware embedded graphics, display, and HIL development, and detailed code generation for embedded systems from AUTOSAR
SCADE Suite	Available	Available	SCADE Suite is a model-based development environment with detailed code generation for safety-critical embedded applications from AUTOSAR
Sim	Available	Available	Visual Integration platform for Software in the Loop from Daimler
SIMULINK	Planned	Available	High and multi-body simulation software from SIMULINK
Simulink	Available	Available	Multi-domain simulation tool for design, analysis and value engineering of complex systems by The MathWorks
Synthesizer	Planned	Planned	Modelica environment from MathWorks
TUC FMI Suite	Available	Available	TUC FMI Suite provides UML and Simulink tools for FMI simulation
TUC FMI Suite	Available	Available	Communication environment from TUC/Thermo
TUC Co-Simulation	Available	Available	Communication tool for FMI plug together modes for Co-simulation
TUC FMI Tool	Available	Available	Co-simulation generation and deployment of model, analysis and value engineering of complex systems by The MathWorks
TUC FMI Tool	Available	Available	Reference FMI tool, supported by manufacturers and volunteers to show how to use FMI with their own tools
UCCO	Available	Available	Reference FMI tool, supported by manufacturers and volunteers to show how to use FMI with their own tools

Basic idea of the joint initiative between MODELISAR and the ASAM XIL group

- ▶ Don't develop competing standards

- Bring together the HIL and MIL/SIL environments
- Project proposal addresses the idea

Several companies, participating in the ITEA-2 project MODELISAR have contacted the ASAM HIL API 1.0 project team meanwhile. It had been evaluated that ASAM HIL API 1.0 functionality could and should also be used for the offline simulation scenarios of the MODELISAR use cases.

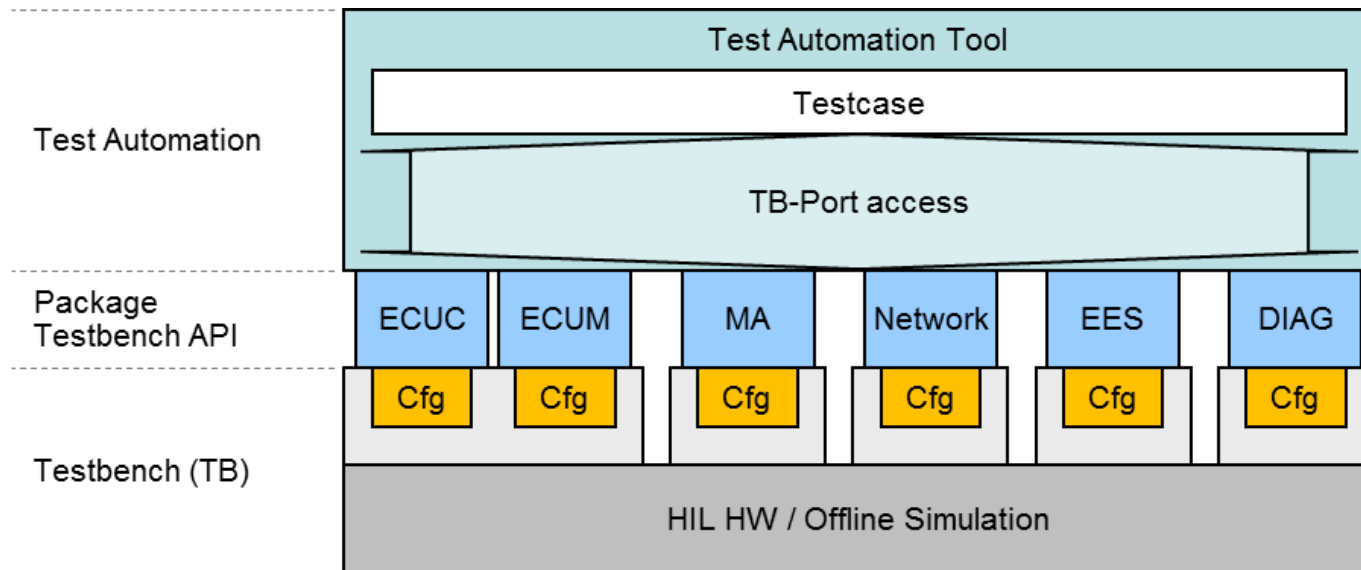
Especially the simulator control API commands, which should be extended in this project will be aligned with already existing MODELISAR results.

taken from project proposal ASAM HIL 2.0.0, 2011

- Daimler entered the ASAM HIL API project to connect the two groups
- No effort on FMI for Application within MODELISAR
- ▶ Common understanding that results concerning offline simulation can be released as FMI for Applications
 - freely available to non-ASAM-members

General Concepts of XIL (1)

Testbench-based Access (as in HIL 1.0.2)

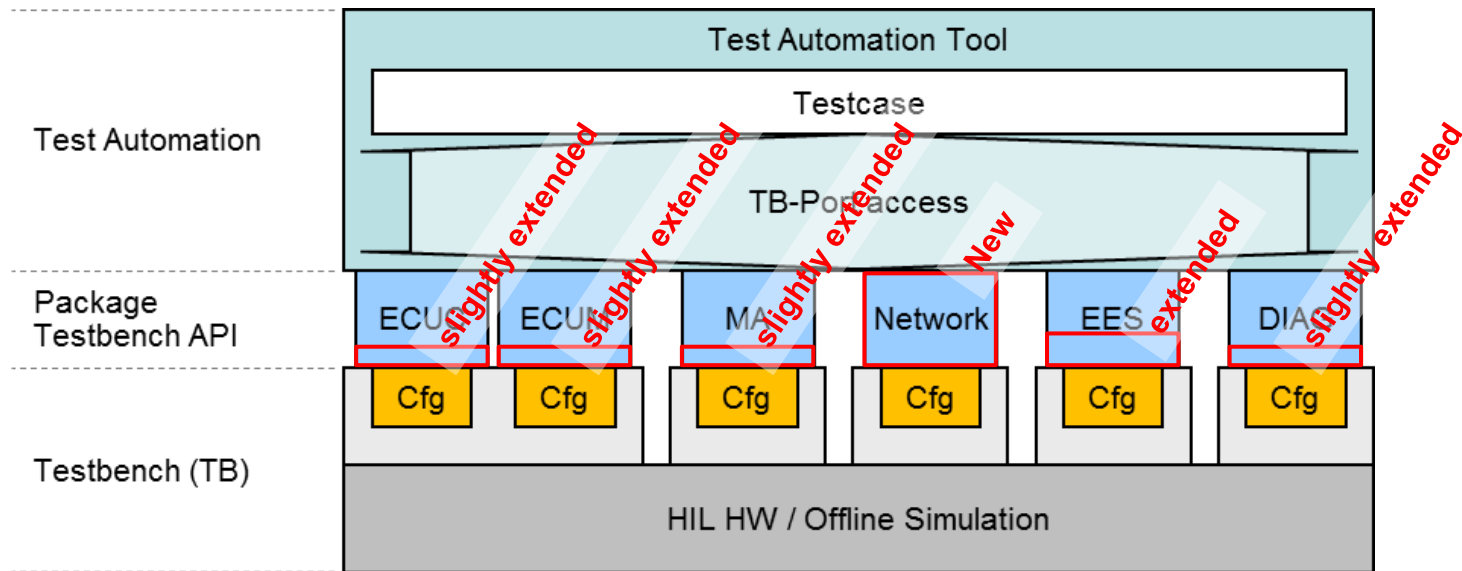


► Drawbacks of HIL 1.0.2:

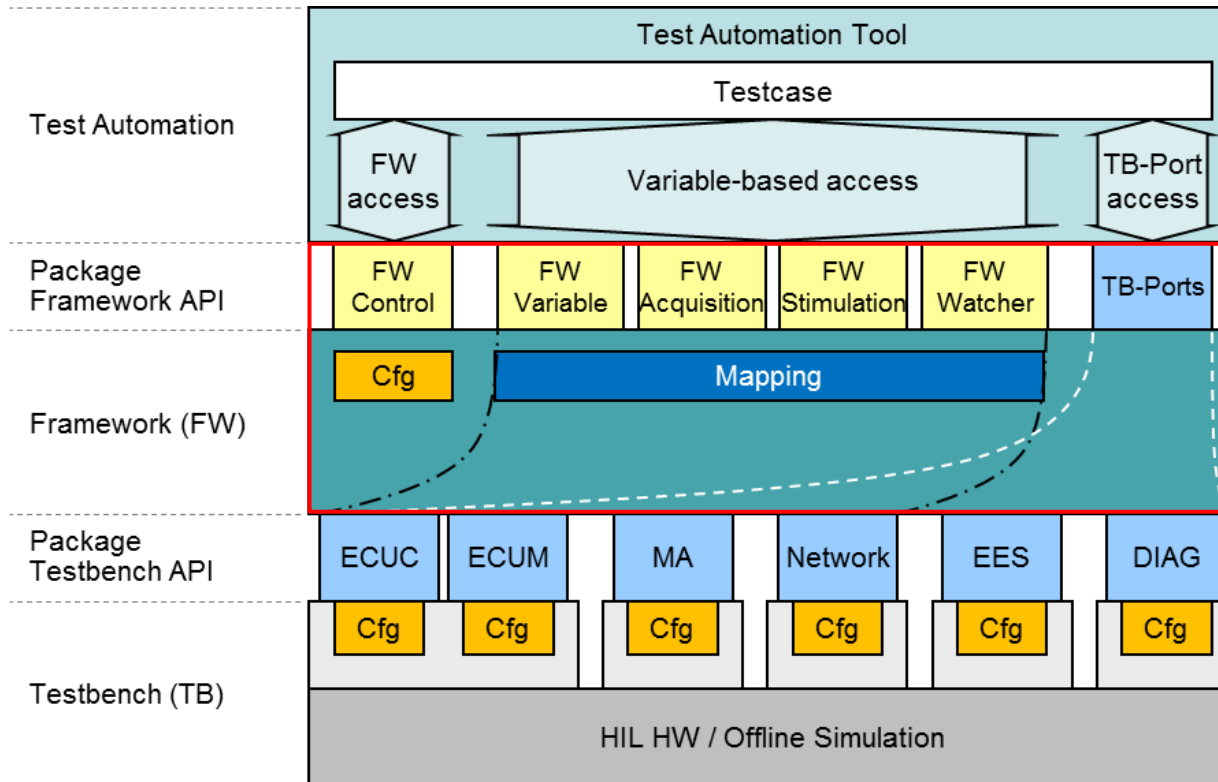
- Testcase has to implement start and shutdown of ports
- Dealing with vendor-specific start and shutdown methods (not standardized in 1.0.2)
- Port-specific variable identifiers and data types
- Missing overall concepts for measuring and stimulating across different ports

General Concepts of XIL (2)

Testbench Extensions with XIL 2.0



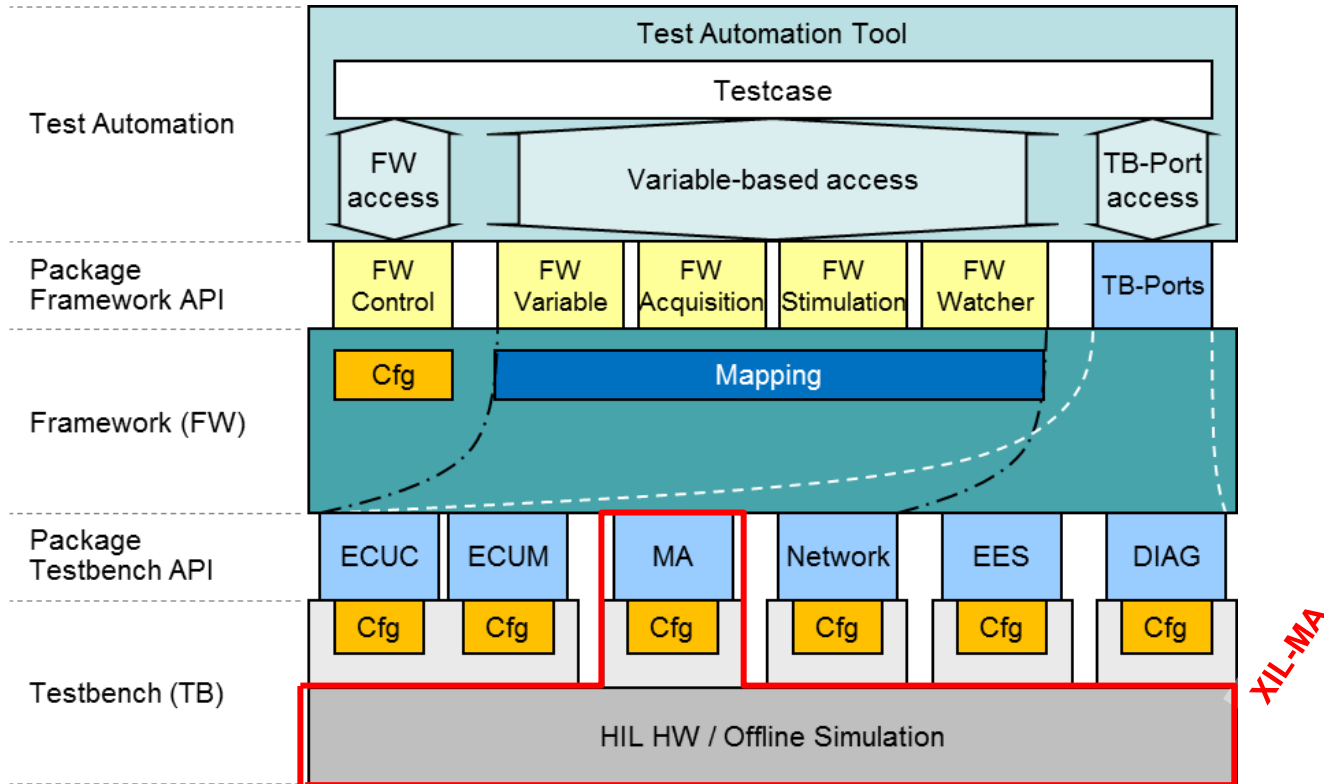
Framework-based Access (with XIL 2.0.0)



Major Benefits:

- Port independence of testcases by using an object-oriented access to variables
- Framework starts and shuts down ports in a configured order
- Test Developer can use both: Testbench Port access and Variable-based access
- FW Variables provide access to the underlying Testbench Port

XIL-MA



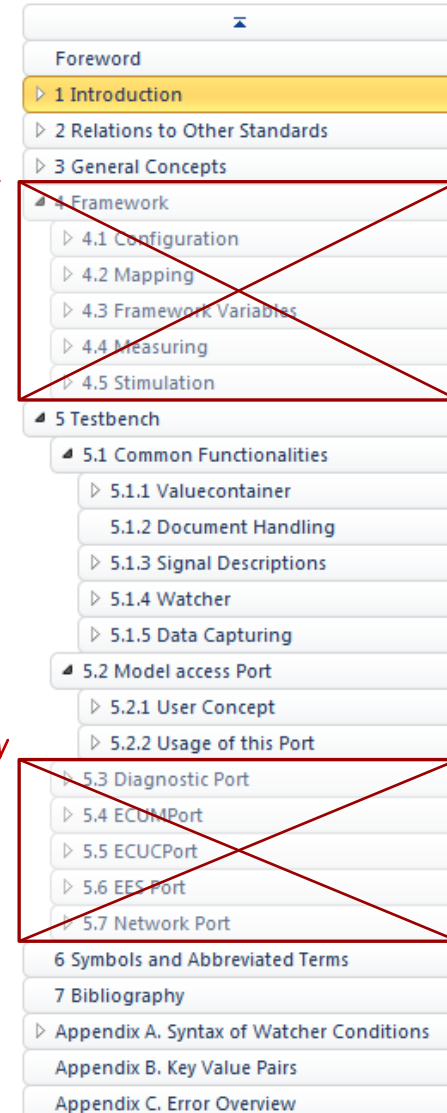
▶ MAPort for Simulation Model Access

- can be used in single testautomation applications
- can be integrated into the XIL Framework together with other ports to benefit from the full Framework functionality.

XIL-MA: What's New?

Not included in XIL-MA but in XIL

- ▶ ASAM releases an additional document:
 - “Generic Simulator Interface for Simulation Model Access”
 - Contains only relevant parts **for implementation of MAPort**
 - **Freely available** for anyone as a download
 - **ASAM XIL standard remains “master”** description, Testbench Model Access Port and Common Functionalities
- ▶ Future releases of ASAM XIL
 - Extract the same scope of content
 - The **FMI group** should be invited to **send representatives** into XIL to contribute their experience (also non ASAM members)



The image shows a table of contents for a document. A red box highlights the '4 Framework' section, which includes sub-sections 4.1 Configuration, 4.2 Mapping, 4.3 Framework Variables, 4.4 Measuring, and 4.5 Stimulation. Another red box highlights the '5.2 Model access Port' section, which includes sub-sections 5.2.1 User Concept and 5.2.2 Usage of this Port. A red arrow points from the text 'Not included in XIL-MA but in XIL' to these two sections. The rest of the table of contents includes: Foreword, 1 Introduction, 2 Relations to Other Standards, 3 General Concepts, 5 Testbench (with sub-sections 5.1 Common Functionalities and 5.3 Diagnostic Port), 6 Symbols and Abbreviated Terms, 7 Bibliography, Appendix A. Syntax of Watcher Conditions, Appendix B. Key Value Pairs, and Appendix C. Error Overview.

- Foreword
- ▶ 1 Introduction
- ▶ 2 Relations to Other Standards
- ▶ 3 General Concepts
- ▶ 4 Framework
 - ▶ 4.1 Configuration
 - ▶ 4.2 Mapping
 - ▶ 4.3 Framework Variables
 - ▶ 4.4 Measuring
 - ▶ 4.5 Stimulation
- ▶ 5 Testbench
 - ▶ 5.1 Common Functionalities
 - ▶ 5.1.1 Valuecontainer
 - ▶ 5.1.2 Document Handling
 - ▶ 5.1.3 Signal Descriptions
 - ▶ 5.1.4 Watcher
 - ▶ 5.1.5 Data Capturing
 - ▶ 5.2 Model access Port
 - ▶ 5.2.1 User Concept
 - ▶ 5.2.2 Usage of this Port
 - ▶ 5.3 Diagnostic Port
- ▶ 5.4 ECUMPort
- ▶ 5.5 ECUCPort
- ▶ 5.6 EES Port
- ▶ 5.7 Network Port
- 6 Symbols and Abbreviated Terms
- 7 Bibliography
- ▶ Appendix A. Syntax of Watcher Conditions
- Appendix B. Key Value Pairs
- Appendix C. Error Overview

Benefits of XIL-MA

- ▶ **Open ASAM Document**
“Generic Simulator Interface for Simulation Model Access
 - Broadens scope of XIL standard to non-ASAM / non-HIL vendors
 - Consistency of specification is maintained by XIL group
 - ASAM ownership and copyright

Compatibility

- ▶ XIL-MA is a subset of XIL.
- ▶ The parts that are available in both standards are absolutely identical.

Deliverables

Package Standard

Directory Specification

ASAM_AE_XIL-MA_AS_V2-0-1.pdf

Directory Generic UML Model

ASAM_AE_XIL-MA_AS_V2-0-1.EAP (view of test case developer)

Package Implementation Support (ASAM software parts)

Directory Templates and Directory Template Example

- Stimulus Signal Description (*.xsd, *.sti and *.stz example)
- ImplementationManifest (ImplementationManifest.xsd; xml example)

Directory Technology Reference Interfaces

Sub Directory Python contains

- Mapping_Rules
(ASAM_AE_XIL_Generic-Simulator-Interface_BS-3-4_Python-API-Technology-Reference-Mapping- Rules_V2-0-1)
- Python Interfaces (py files)

Sub Directory C# contains

- Mapping_Rules (ASAM_AE_XIL_Generic-Simulator-Interface_BS-2-4_C#-API-Technology-Reference-Mapping-Rules_V2-0-1)
- Interfaces (cs files)
- Sample Code (restricted for MA-Port)

Changes in Maintenance XIL-MA 2.0.1

based on XIL-MA 1.0.0

- ▶ Bugfixing of some workblockers, that have been detected during implementation phase, such as scalar was returned instead of a list of scalars;
Signal generator factory now returns the correct ISignalGeneratorSTZWriter instead of ISignalGeneratorSTZReader;
added missing value 'eDATAFILE' to SegmentTypes enum.
- ▶ Definition of Initial values to avoid invalid object creation of the new DataFileSegment class.
- ▶ Added some missing error codes and post conditions
- ▶ Added a new MAPort method GetTaskInfos to get information about existing tasks (eTimerDriven, eEventDriven, Sample Period)
- ▶ Added some functionality for simultaneous read access of multiple clients and threads to the Testbench Manifest File (contains vendor-specific information about the Testbench)
- ▶ Capture now derives from Interface IDisposable to enable explicit instance destruction of Capture (to free system resources, e. g. real-time service code)
- ▶ Correction of errors in documentation (Guide and UML Model, e. g. SetStartTriggerCondition)
- ▶ Introduction of the correct short name in all documents:
"XIL - Generic Simulator Interface" instead of "XIL - API for ECU Testing via XIL"

Changes in Maintenance XIL-MA 2.0.2

based on XIL-MA 2.0.1

Data Capturing (motivated by MAPort)

- ▶ VariableInfoObject now contains AvailableTasks
- ▶ Trigger Time Stamps added to Capture Result (eSTARTTRIGGER, eSTOPTRIGGER)
- ▶ Data Frame Time Stamps added to Capture Result (eDATAFRAMESTART, eDATAFRAMESTOP)

Improvements & Convenience

- ▶ Various improvements in documentation and examples (e. g. Trigger with start and stop delay)
- ▶ Missing factory method added (SignalGeneratorFactory.CreateSignalGenerator)
- ▶ Missing Post Conditions added (for better error handling)