Functional Mockup Interface
The FMI standard for model exchange

Jakob Mauss, Andreas Junghanns
QTronic GmbH

January 2010

Outline

- motivation for standardized models
- key requirements
- structure of the model interface
  - model execution
  - model description
- FMI tools
- summary

presentation based on working results of Modelisar WP200

A Motivation

QTronic
SIMULATION FOR ENGINEERING

AUTOSAR

Tool for developing automotive control software

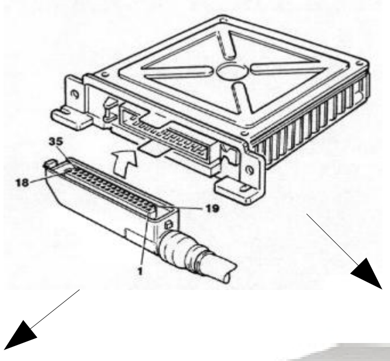
MODELICA

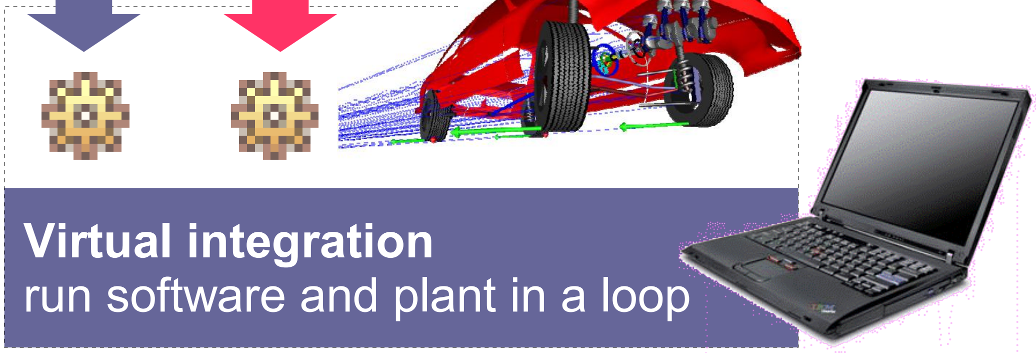
Simulation tool for developing plant model

push-button solution thanks to standard

missing standard

?





Virtual integration
run software and plant in a loop

SiL/MiL

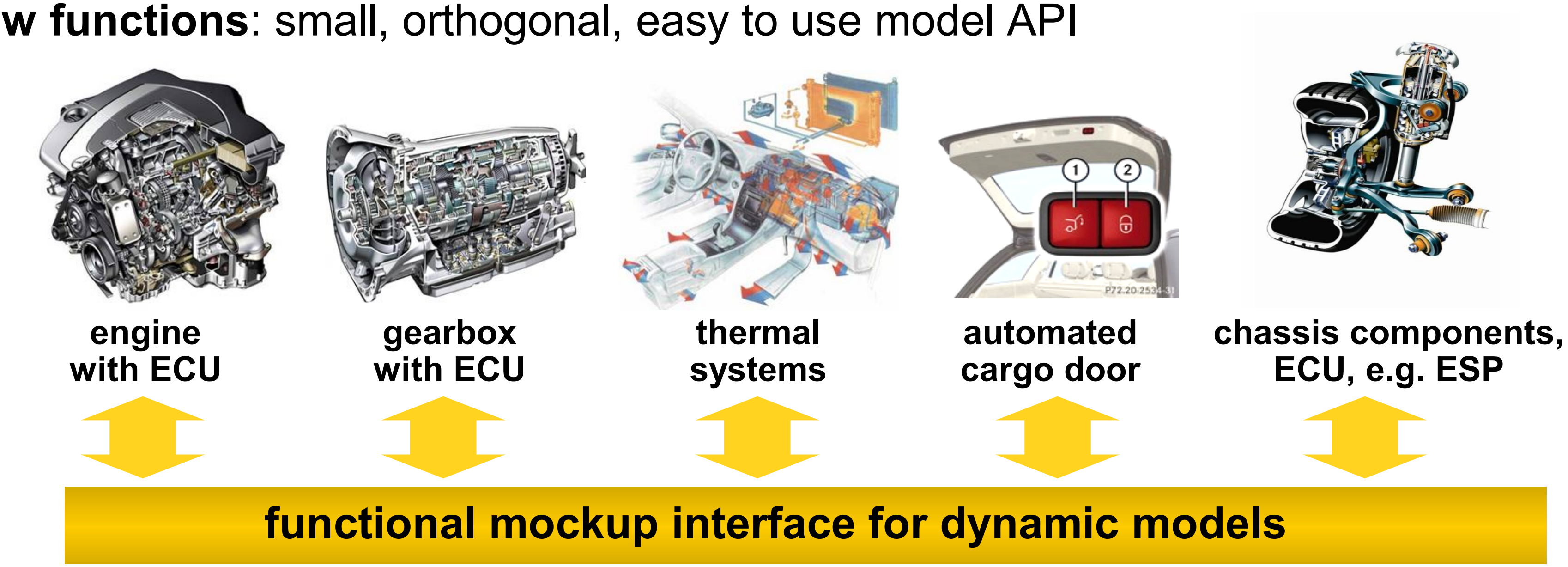
Prototype

HiL

- Autosar enables push-button solution for running automotive software on a laptop (SiL/MiL)
- this will change the economy of simulation in the automotive development process
- push button solution for simulation needed: The Modelisar exchange format for models

Key requirements, from FMI specification

- **Expressivity**: cover at least Modelica, Simulink S-Function, SIMPACK
- **Large models**: up to 10^4 states, 10^6 variables
- **Simulator and Processor independence**: target-independent model exchange format
- **Minimize execution time**: minimize model - simulator communication
- **Multiple instances**: support many instances of the same model
- **Many and nested models**: a model may contain models
- **Small memory footprint**: support models running on ECU
- **Few functions**: small, orthogonal, easy to use model API



engine with ECU

gearbox with ECU

thermal systems

automated cargo door

chassis components, ECU, e.g. ESP

functional mockup interface for dynamic models

Structure of the model interface

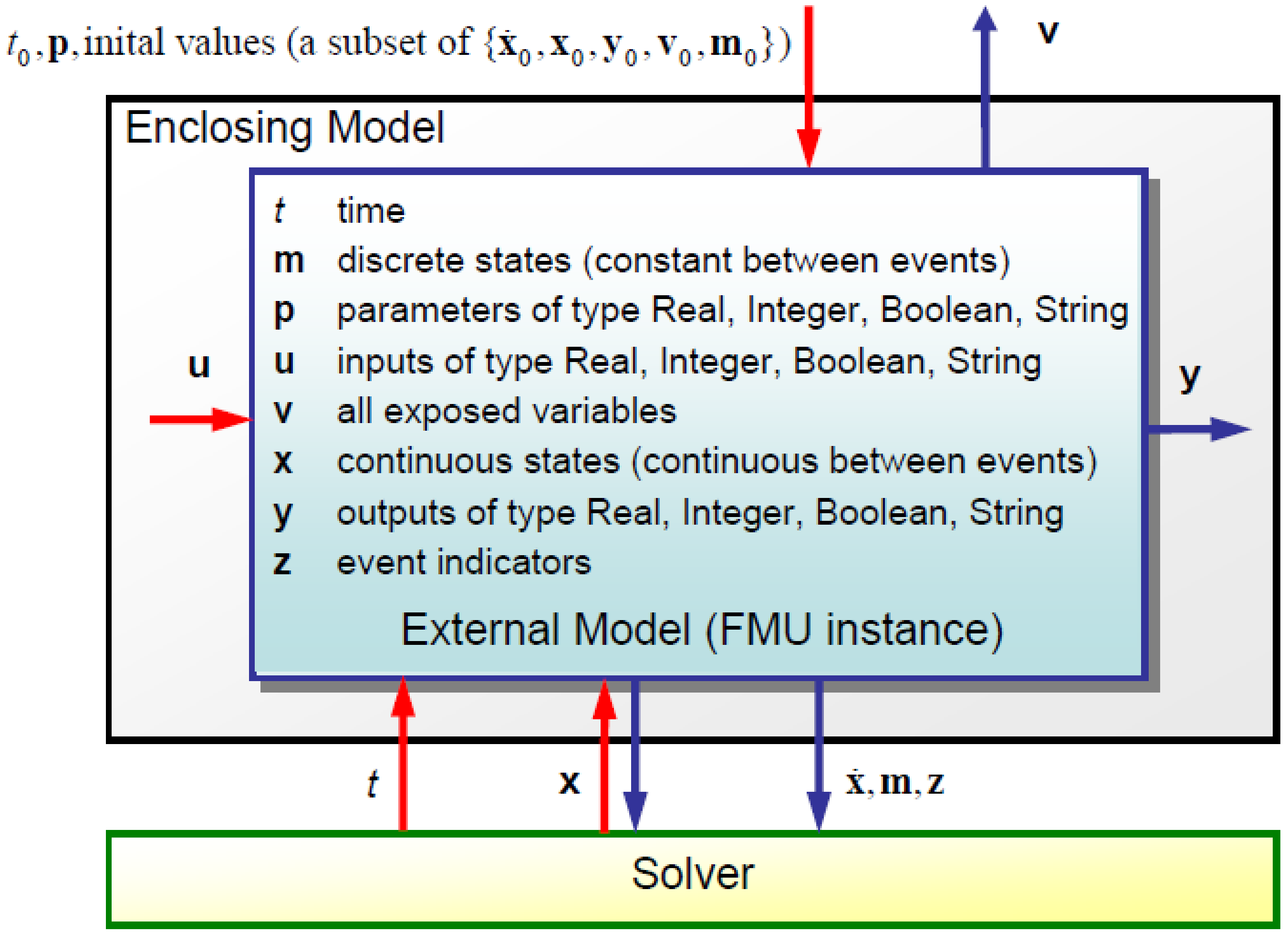
The FMI specification defines

- Model execution interface: API for simulating a model
- Model description: Info about all variables as XML, mostly needed by GUI
  - XML offers more flexibility than a C API, e.g. for processing from Java
  - Separation of symbol table and executable leads to small executable, good for models that are executed by an ECU

Models are exchanged as zip file with suffix .fmu containing

- executable DLL or C source code
- model description as XML file

Model execution interface

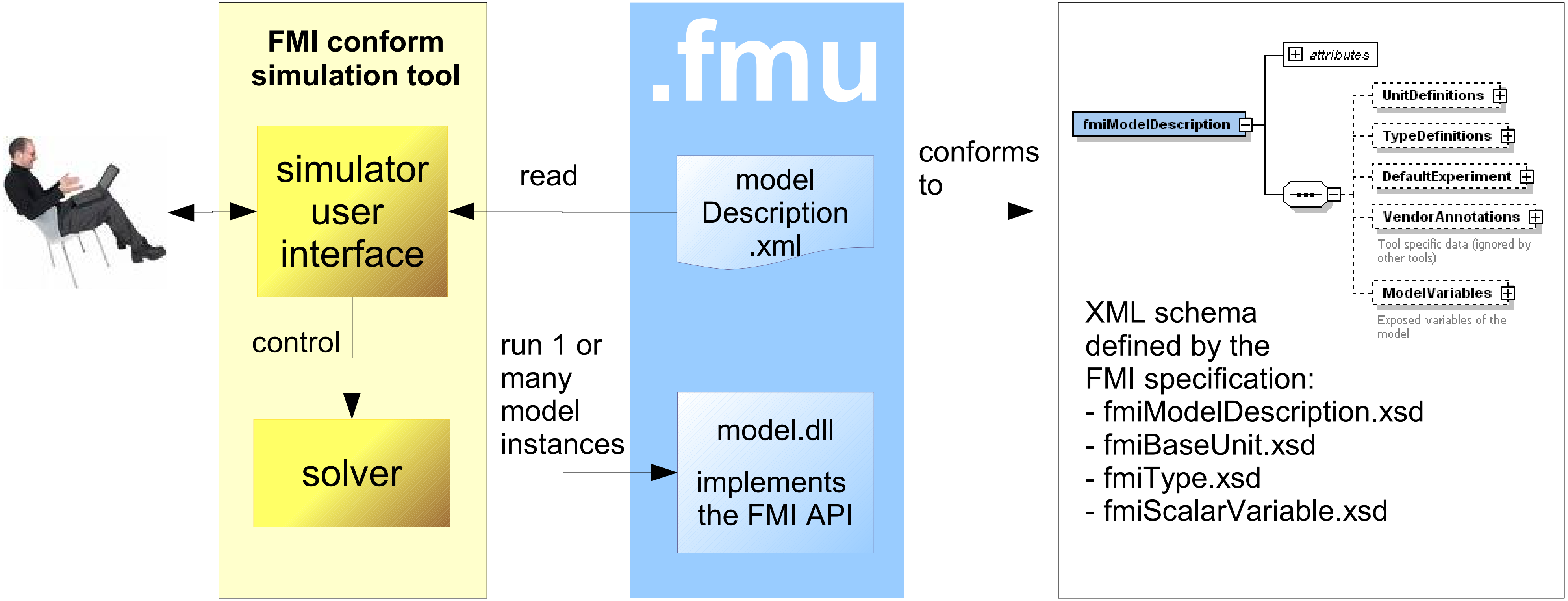

$t_0, \mathbf{p}$, inital values (a subset of $\{\dot{\mathbf{x}}_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{v}_0, \mathbf{m}_0\}$)
v
Enclosing Model
t time
m discrete states (constant between events)
p parameters of type Real, Integer, Boolean, String
u inputs of type Real, Integer, Boolean, String
v all exposed variables
x continuous states (continuous between events)
y outputs of type Real, Integer, Boolean, String
z event indicators
External Model (FMU instance)
u
y
t
x
$\dot{\mathbf{x}}, \mathbf{m}, \mathbf{z}$
Solver

model shown here
for the case of an ODE

not shown: support for
- self-integrating models that include a solver
- DAE
- analytic Jacobians
- direct feed through

for more details, see the FMI specification: http://www.functional-mockup-interface.org/

Model description

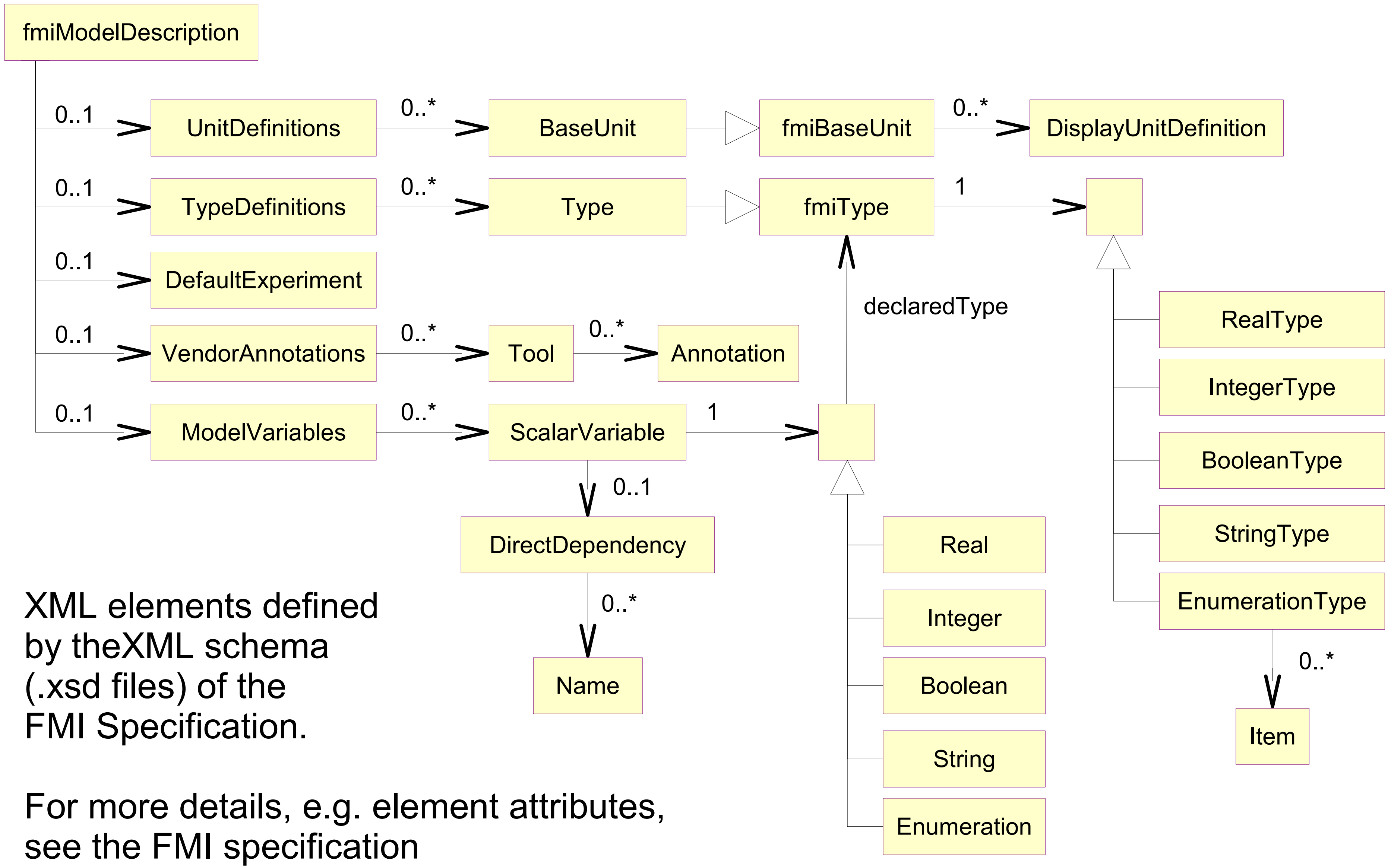


XML schema defined by the FMI specification:

- fmiModelDescription.xsd
- fmiBaseUnit.xsd
- fmiType.xsd
- fmiScalarVariable.xsd

For more details, see the FMI specification: http://www.functional-mockup-interface.org/

Model description



XML elements defined by theXML schema (.xsd files) of the FMI Specification.

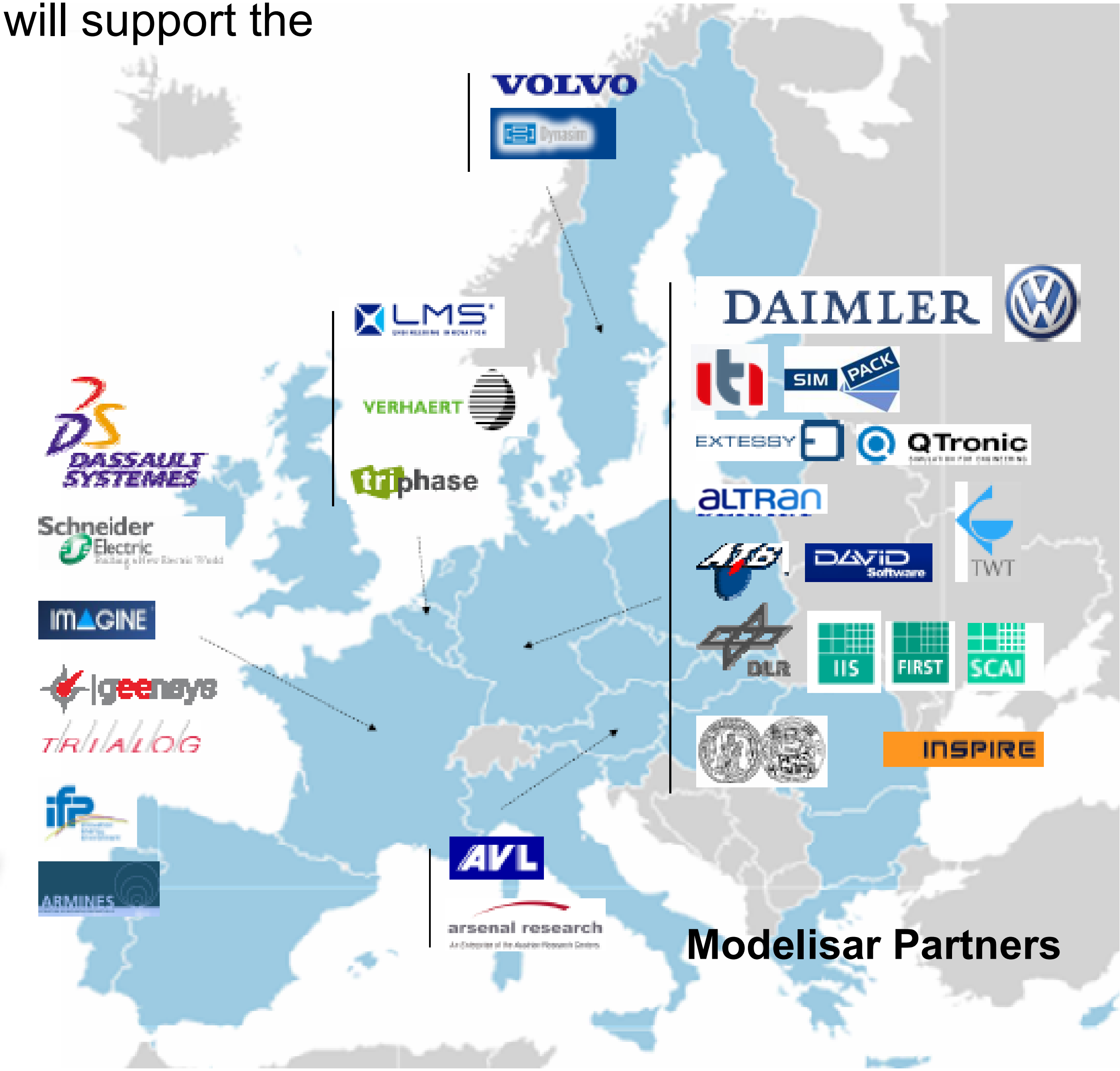
For more details, e.g. element attributes, see the FMI specification http://www.functional-mockup-interface.org/

Tools supporting the FMI standard

The following simulation tools will support the FMI standard in 2010

- AMESim
- Dymola 7.4
- Silver 2.0
- SimulationX
- SIMPACK

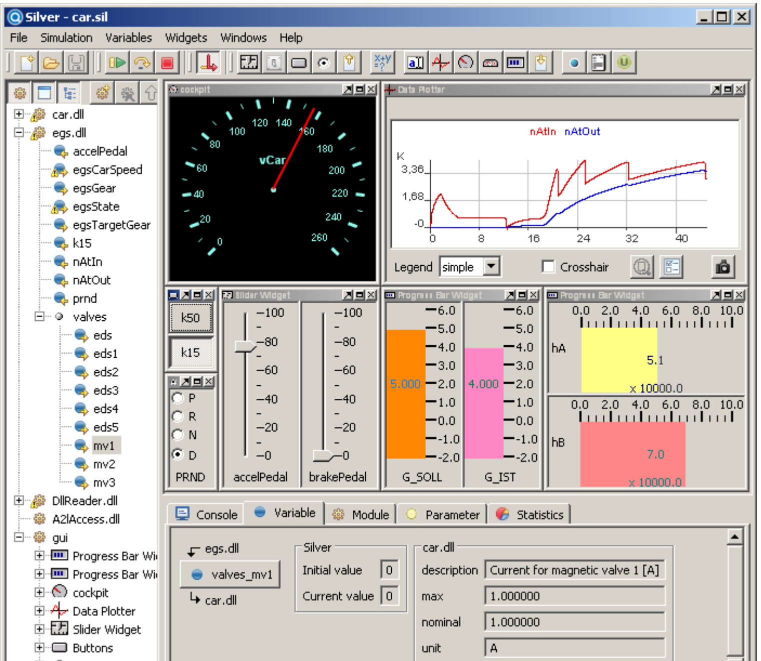
The FMI specification is developed within the ITEA-2 project Modelisar 2008 - 2011



Modelisar Partners

Silver 2.0 will implement the FMI

- Silver 2.0 runs FMI conform rmodels
- key features
  - self-configuring: no wiring needed
  - models are self-integrating or use solvers provided by Silver
  - configurable user interface to control and visualize a simulation
  - debugging: stepper, breakpoints, pdb
  - special support for automotive software a2l connection, xcp emulation, read/write mdf, dcm, hex, ...


Silver - car.sil

File Simulation Variables Widgets Windows Help

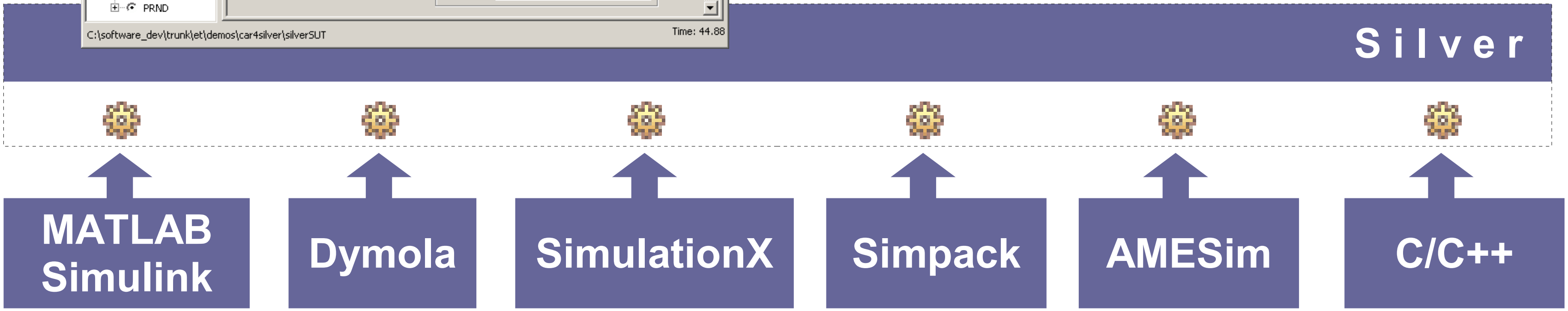
Silver: Initial value 0, Current value 0

car.dll

description	Current for magnetic valve 1 [A]
max	1.000000
nominal	1.000000
unit	A

C:\software_dev\trunk\et\demos\car4silver\silverSUT

Time: 44.88



Summary

- FMI defines an exchange format for hybrid ODE/DAE models, without (self-integrating) or including a numerical solver

- FMI model is zip file containing
  - DLL (to protect IP) and/or the model's C source
  - XML file describing the model, e.g. its variables

- FMI conform models generated by tools such as:
  AMESim, Dymola, Simpack, SimulationX
  wrapping of MATLAB/Simulink S-functions possible

- FMI specification
  - available for free from: http://www.functional-mockup-interface.org/
  - validated using prototype implementations from various tool vendors

- FMI is expected to boost the use of simulation-based development (SiL/MiL) of automotive software