ITEA Office
High Tech Campus 69 - 3
5656 AG Eindhoven
The Netherlands

T + 31 88 003 6136
E info@itea3.org
W www.itea3.org

ITEA 3 is a EUREKA strategic ICT cluster programme

Exploitable Results by Third Parties

14018 OpenCPS

Project details

Project leader:	Dr. Magnus Eek
Email:	magnus.eek@saabgroup.com
Website:	www.opencps.eu

Name: OMSimulator		
Input(s):	Main feature(s)	Output(s):
▪ Functional Mock-up Units (FMUs) ▪ and/or System Structure & Parameterizations (SSPs) ▪ and/or simulation models in third party tools	▪ Open source co-simulation tool for standardized, numerically robust, and efficient distributed simulation ▪ Supporting i) Functional Mock-up Interface (FMI) 2.0 CS & ME, ii) System, Structure & Parameterization (SSP), and iii) the Transmission Line element Method (TLM) ▪ Supports integration of FMUs as well as external tool integration, e.g. Simulink, Adams, Hopsan, Dymola	▪ Simulation results ▪ Composite model definition according to SSP

Unique Selling Proposition(s):	▪ Industry-grade and open-source model integration and co-simulation tool ▪ Combines the benefits of FMI, SSP, and TLM ▪ Integrated with GUI support in OpenModelica and Papyrus, and available as standalone for integration into scripting frameworks, third-party tools, and specialized applications
Integration constraint(s):	▪ Available for Windows and Linux platforms
Intended user(s):	▪ Industry and academia
Provider:	▪ Open Source Modelica Consortium (OSMC)
Contact point:	▪ Prof. Peter Fritzson (Linköping University)
Condition(s) for reuse:	▪ Open-source conditions according to OSMC: https://openmodelica.org/home/consortium

Latest update: 16 November 2018

Name: PhiSystem / Simulation module		
Input(s):	Main feature(s)	Output(s):
▪ Functional Mock-up Units (FMUs) ▪ OMSimulator ▪ System architecture description (SysML) developed with the initial version of PhiSystem	▪ Simulation architecture definition ▪ Traceability between system and simulation architecture ▪ Subsystem model specification (purpose and I/O interface) ▪ FMUs integration ▪ Simulation execution and analysis	▪ Simulation results ▪ Link with system objectives and requirements via metrics assessment

Unique Selling Proposition(s):	▪ Available as standalone with PhiSystem ▪ Open-source license
Integration constraint(s):	▪ Availability of OMSimulator
Intended user(s):	▪ Industry and academia ▪ In industry: system architects and/or system designers
Provider:	▪ Sherpa Engineering (www.sherpa-eng.com)
Contact point:	▪ Philippe Fiani (p.fiani@sherpa-eng.com)
Condition(s) for reuse:	▪ Open-source conditions

Latest update: 19 November 2018

Name: FMI Export module in Pro-SiVIC		
Input(s):	Main feature(s)	Output(s):
▪ Autonomous driving scenario ▪ 3D scene ▪ Sensing configuration	▪ Selection of entities (sensors) to be exported ▪ Export of all perception sensors' data interface in an FMU ▪ Supporting Functional Mock-up Interface (FMI) 2.0 CS	▪ FMUs exposing Sensor Data outputs
Unique Selling Proposition(s):	▪ Availability of sensor's raw output to address perception performance issues ▪ Simulation can be executed on different machine than FMI model	
Integration constraint(s):	▪ Windows platform.	
Intended user(s):	▪ Car Manufacturers, Tier One and Tier Two suppliers developing or integrating Advanced Driver Assistance Systems (ADAS) and/or Autonomous Driving (AD) functions	
Provider:	▪ ESI Group	
Contact point:	▪ Rodolphe.Tchalekian@esi-group.com (EMEA Pro-SiVIC Business Developer) ▪ Philippe.DeSouza@esi-group.com (Pro-SiVIC Product Manager)	
Condition(s) for reuse:	▪ Commercial Licenses (Yearly or Paid-up) ▪ Research Licenses (Yearly or Paid-up)	
	Latest update: 20 November 2018	

Name: txtUML		
Input(s):	Main feature(s)	Output(s):
▪ Textual Model Code ▪ Deployment Configuration ▪ FMU Configuration	▪ Generation of UML2 model from textual model supporting wide variety of standard UML2 elements ▪ Automatic parallelization of classes based on deployment configuration ▪ Generation of standard FMU based on FMI configuration	▪ Model Adaptation in C++ ▪ Generated FMU environment ▪ FMU packaged according to standard
Unique Selling Proposition(s):	▪ Textual (VCS friendly) executable UML model ▪ Automatic generation of UML diagrams ▪ Live debug and testing through state machine and sequence diagrams	
Integration constraint(s):	▪ Available for all platforms inside Eclipse ecosystem	
Intended user(s):	▪ Industry and academia	
Provider:	▪ ELTE-Soft, Eötvös Loránd Science University, Budapest	
Contact point:	▪ Gera Zoltán, ELTE University: https://github.com/ELTE-Soft/txtUML	
Condition(s) for reuse:	▪ Open-source with EPL-1.0 license: https://github.com/ELTE-Soft/txtUML/blob/master/LICENSE	
	Latest update: 16 November 2018	

Name: xtUML-Wrapper for FMI 2.0		
Input(s):	Main feature(s)	Output(s):
▪ Functional Mock-up units (FMUs) designed and realized using Model Based Systems Engineering (MBSE) methodology ▪ Model and interface configuration files ▪ Simulation infrastructure that is compliant with the FMI 2.0 standard	▪ Provides a generic MBSE (xtUML) model of the FMI 2.0 standard that enables straight forward integration of non-FMI compliant MBSE-modelled simulation entities ▪ Integrates with any FMI-compliant simulation master ▪ Uses the xtUML methodology and tool set for modeling and generation of executables ▪ Provides the first working MBSE-model of the FMI 2.0 standard API	▪ An functional MBSE-model integrated in a running simulation environment ▪ Simulation results

Unique Selling Proposition(s):	▪ Provides integration of any xtUML-model with a simulation infrastructure supporting the FMI 2.0 standard API. ▪ Combines the benefit of FMI and MBSE (xtUML)
Integration constraint(s):	▪ Available for Windows and Linux platforms
Intended user(s):	▪ Industry and academia
Provider:	▪ Saab AB
Contact point:	▪ Ph.D. Nils Paulsson
Condition(s) for reuse:	▪ Commercial license to be negotiated, a free license can be provided for research purposes

Latest update: 22 November 2018

Name: SSP-FMU design extension for Papyrus		
Input(s):	Main feature(s)	Output(s):
▪ Functional Mock-up Units (FMUs) ▪ and/or System Structure & Parameterizations (SSPs). ▪ OMSimulator for executing simulation.	▪ Open-source extension for Papyrus supporting the definition of complex simulation scenarios. ▪ Authoring tool for SSP descriptions that provides direct synchronization between SysML bbds and SSP. ▪ Scripting platform for defining multiple simulation scenarios and advanced visualization.	▪ Composite model definition according to current SSP standard. ▪ Simulation results

Unique Selling Proposition(s):	▪ Supporting round-trip engineering between SysML and SSP for architecture and simulation design ▪ High-level scripting support for defining complex simulation scenarios ▪ Advanced browser based result visualization based on Jupyter's notebook mechanism.
Integration constraint(s):	▪ Available on all platforms supporting Java
Intended user(s):	▪ Industry and academia
Provider:	▪ CEA and IncQuery Labs as part of Eclipse Modeling Project
Contact point:	▪ Sebastien Revol (CEA)
Condition(s) for reuse:	▪ Open source under EPL v2.0: https://www.eclipse.org/legal/epl-v20.html

Latest update: 29 November 2018

Name: Papyrus support of OMG Precise Semantics for State Machine (PSSM) standard		
Input(s):	Main feature(s)	Output(s):
▪ UML models containing State Machines complying to PSSM OMG standard	▪ Open Source simulation engine interpreting the state machines, fully aligned with OMG standard specifications ▪ Diagram animation support ▪ Debug support	▪ Simulation results

Unique Selling Proposition(s):	▪ Only Open Source tool providing a full support of the Object Management Group standard specification for model execution
Integration constraint(s):	▪ Available on all platforms supporting Java
Intended user(s):	▪ Industry and academia
Provider:	▪ CEA as part of Eclipse Modeling Project
Contact point:	▪ Jeremie Tatibouet (CEA)
Condition(s) for reuse:	▪ Open source under EPL v2.0: https://www.eclipse.org/legal/epl-v20.html

Latest update: 29 November 2018

Name: Multi-Domain Modelica Models for Gas Turbine and Power Grid Analysis		
Input(s):	Main feature(s)	Output(s):
▪ Model parameters	▪ Modelica package for turbo-mechanical gas turbine models. ▪ Modelica package for power systems models ▪ Modelica package with examples and interfaces for multi-domain models	▪ Simulation results
Unique Selling Propositio n(s):	▪ Gas turbine models – based on the ThermoPower library – are adapted to be used in combination with power system models, based on the OpenIPSL library. ▪ Suitable dynamic study of detailed network models with electrical and mechanical components. ▪ The library can be extended to other domains, eliciting the representation of power systems components with further details.	
Integratio n constraint (s):	▪ Dymola and OpenModelica (partial support)	
Intended user(s):	▪ Industry and academia	
Provider:	▪ Royal Institute of Technology (KTH) ▪ Rensselaer Polytechnic Institute (RPI) ▪ Instituto Costarricense de Electricidad (ICE)	
Contact point:	▪ Luigi Vanfretti (RPI), Miguel Aguilera (ICE) via: ▪ https://github.com/ALSETLab/2018_AmericanModelicaConf_PowerGrid_plus_PowerSystems	
Condition(s) for reuse:	▪ Licensed under the GNU GPL v3 ▪ See terms under: https://github.com/ALSETLab/2018_AmericanModelicaConf_PowerGrid_plus_PowerSystems/blob/master/LICENSE	
	Latest update: 30 November 2018	

Name: IDA Software-in-the-loop simulator for PLC code		
Input(s):	Main feature(s)	Output(s):
▪ IEC 61131-3 PLC control code from the open source Beremiz authoring system ▪ Connections between the controller(s) and the physical system model	▪ Fast simulation of IEC 61131-3 PLC code coupled with IDA based simulators ▪ Control code can be simulated several orders of magnitude faster than with HIL testing (which is also available) ▪ Both complex MIMO controllers and elementary function blocks can be imported	▪ Simulation results ▪ Controller export according to IEC 61131-10 (PLCOpen XML) is planned

Unique Selling Proposition(s):	▪ Enables industrial controllers in IDA based physical system models for energy systems, buildings, and tunnels ▪ No manual interpretation of simulated control behavior has to be done for industrial implementation of developed controller since PLC control code can be used directly in the simulator – less error-prone development process
Integration constraint(s):	▪ Available for Windows ▪ Requires the open source Beremiz IDE for IEC 61131-3 model authoring
Intended user(s):	▪ Industry
Provider:	▪ EQUA Simulation AB
Contact point:	▪ Per Sahlin (EQUA)
Condition(s) for reuse:	▪ Commercial license

Latest update: 3 December 2018