The research leading to these results has received funding from the European Union’s Seventh Framework Programme (FP7/2007-2013) for CRYSTAL – Critical System Engineering Acceleration Joint Undertaking under grant agreement n° 332830 and from specific national programs and / or funding authorities.
CRYSTAL facts and figures

- 70 Partners
- 10 Countries
- Automotive, Aerospace, Healthcare, Rail
- € 82 Mio
- 2013-2016
- Coordinated by AVL
Motivation for CRYSTAL

- The development of modern embedded systems typically requires a **large number of tools** from different software vendors and often custom in-house solutions.
- Tool **landscapes have traditionally emerged in an evolutionary process** from single tools that were never designed to collaborate.
- Developers and engineers are now often forced to **manually transfer information from one tool to the other**, which entails a number of problems:
  - Increased development time and cost
  - Late changes are extremely costly
  - High risk of inconsistencies
  - Safety may be compromised
Motivation for CRYSTAL cont’d

- **Tool integration** often established in an ad-hoc manner by creating proprietary bridges between each pair of tools
  - Not scalable (exponential effort)
  - Vulnerable to version upgrades and process changes

- The current situation *does not support efficient collaboration throughout the product lifecycle* and across vendor boundaries
Objectives of CRYSTAL

- Establish an **Interoperability Specification (IOS)** and a **Reference Technology Platform (RTP)** that allows loosely coupled tools to share and interlink their data across the **entire product lifecycle** of a safety-critical embedded system.

- The big vision is that the CRYSTAL IOS and RTP will become a **“de facto” standard** for enabling interoperability for future embedded systems engineering.
Classical automotive V-Model

Requirements Specification

Vehicle conceptualization (Engine, Powertrain and Chassis)

Module conceptualization (Engine, Powertrain and Chassis)

Module test (Engine, Powertrain, Chassis)

Implementation

Building a vehicle

Integration test

System Validation & Verification

Common challenges:
- Establish traceability between all data, e.g.
  - for change impact analysis, grouping parameters
  - test runs and measurement results
- Put all data under versioning control

Involved Artifacts:
- Requirements
- Formal Requirements
- Parameters
  - for UUT
  - for Testbed
  - for Environment
- Test-Runs
- Models
- Measurement Results
- Validation Results
Common challenges:
- Establish traceability between all data, e.g.
  - for change impact analysis, grouping parameters
  - test runs and measurement results
- Put all data under versioning control

Involved Artifacts:
- Requirements
- Formal Requirements
- Parameters
  - for UUT
  - for Testbed
  - for Environment
- Test-Runs
- Models
- Measurement Results
- Validation Results
Multiple Interoperability Standards?

- **OSLC**
  - connect artifacts (PLM)

- **STEP**
  - exchange PDM data

- **ASAM ODS, MCD2**
  - exchange Meas.Data, Calibr.Data,...

- **AUTOSAR Software-Architecture**
  - exchange SW-models

- **FMI**
  - Connect simulation models

---

2013-12-03.-04 ASAM International Conference Dresden

Horst Pfluegl / AVL
Multiple Interoperability Standards?

- Interoperability can be only defined with respect to a given challenge (interoperability challenge):
  - simple traceability between artifacts
  - change – impact analysis
  - semantic-preserving transformation
  - export/import of huge data sets
  - reuse of parameters or models
  - heterogeneous co-simulation
  - etc.
Automotive Example 1
Traceability Requirements <-> Simulation Models <-> Test Cases

Integration of Authoring Tools in ALM Environments
Proposed Approach (Example 1)

IOS / OSLC
Shared Artifacts for Integration

Model

Element

Element

Project

TC

RQ

OSLC AM

OSLC RM

OSLC QM

Generic Semantics

Other models and standards

Meta-Model A

Authoring Tool (e.g., Simulink)

Proprietary Representation

ALM Environment

Horst Pfluegl / AVL

2013-12-03-04 ASAM International Conference Dresden
Automotive Example 2
Co-simulation of heterogeneous models

- Compose virtual vehicles out of heterogeneous models
- Global **data and configuration management**
- Perform **co-simulation**
Proposed Approach (Example 2)

Uniform Workbench

IOS / OSCL
Shared Artifacts for Integration

OSLC AM

Model
Element
Element

OSLC AM

Model
Element
Element

OSLC Adaptor

OSLC Adaptor

Other models and standards

Meta-Model A
Vehicle Simulation

Co-Simulation Execution (e.g., via FMI)

Meta-Model B
Engine Simulation

Generic Semantics
Specific Semantics

2013-12-03-04 ASAM International Conference Dresden

Horst Pfluegl / AVL
Automotive Example 3
Engine calibration process

- **Traceability** between requirements, test/calibration set-up, calibration data, measurement results
- **Change-Impact analysis**
- **Configuration, version and variant management**
- **Import / export** of calibration data
- etc.
Proposed Approach (Example 3)

IOS / OSLC

Shared Artifacts for Integration

- Project
- Calibration DataSet
- Calibration SetUp
- Param
- Model
- Test Series
- Testrun

OSLC Adaptor

ASAM MCD-2 MC

Calibration Data Management (CDM)

Export Calibration Data

Proprietary Representation

Export Measurement Data

ASAM ODS

Measurement Result Database
CRYSTAL has a strong focus on OSLC
  - dedicated to interoperability for life-cycle management
  - bridge from product engineering to system engineering

CRYSTAL will integrate also other interoperability standards
  - different standards for different purposes
  - e.g., link heterogeneous models with OSLC but perform co-simulation via Functional Mock-Up Interface (FMI)
  - e.g., provide a link to a set of calibration data with OSLC, but define the finer syntactic structure and semantics with ASAM MCD-2 MC

CRYSTAL will draw an integral picture 😊