Modelica Association
System Structure and Parameterization (SSP)
Project Overview

SMSWG Meeting November 29, 2016

Hubertus Tummescheit

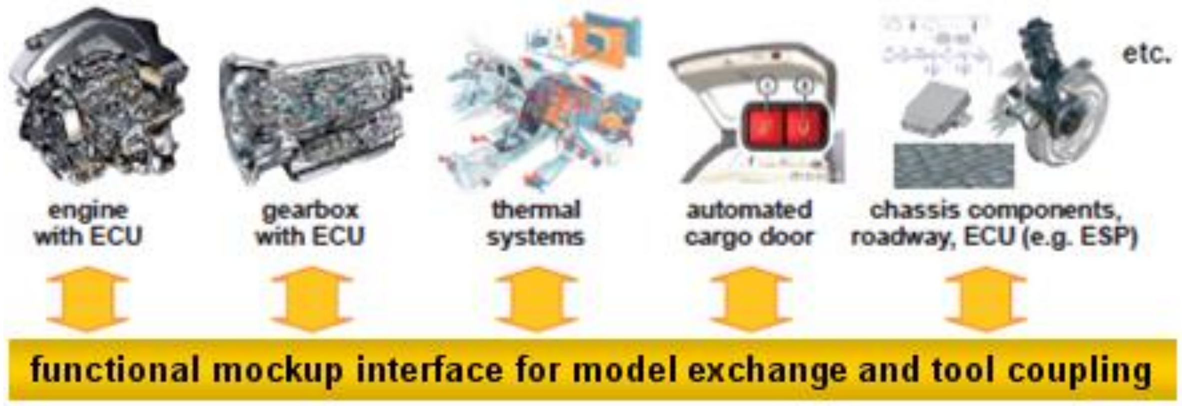
CEO Modelon Inc. & Board member Modelica Association

Acknowledgements: Members of the Modelica Association SSP working group

M. Deppe — J. Köhler — J. Krasser — P. R. Mai — M. Nagasawa

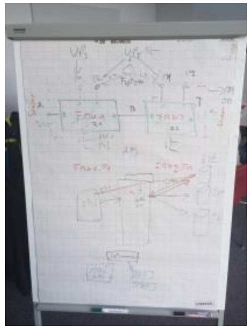
Motivation for initiating MAP "System Structure and Parameterization" (SSP) – Using FMI as Basis

- FMI is basically a great technology to make exchanging models inside and among companies much easier
- Typical use-case is a network of FMUs (System structure) …



- … Therefore, some features are missing …

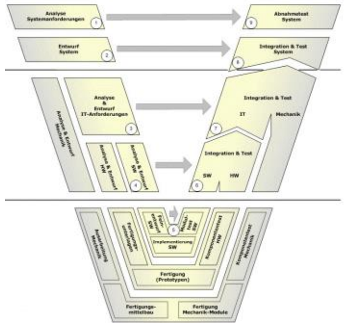
Motivation for initiating SSP – Missing features



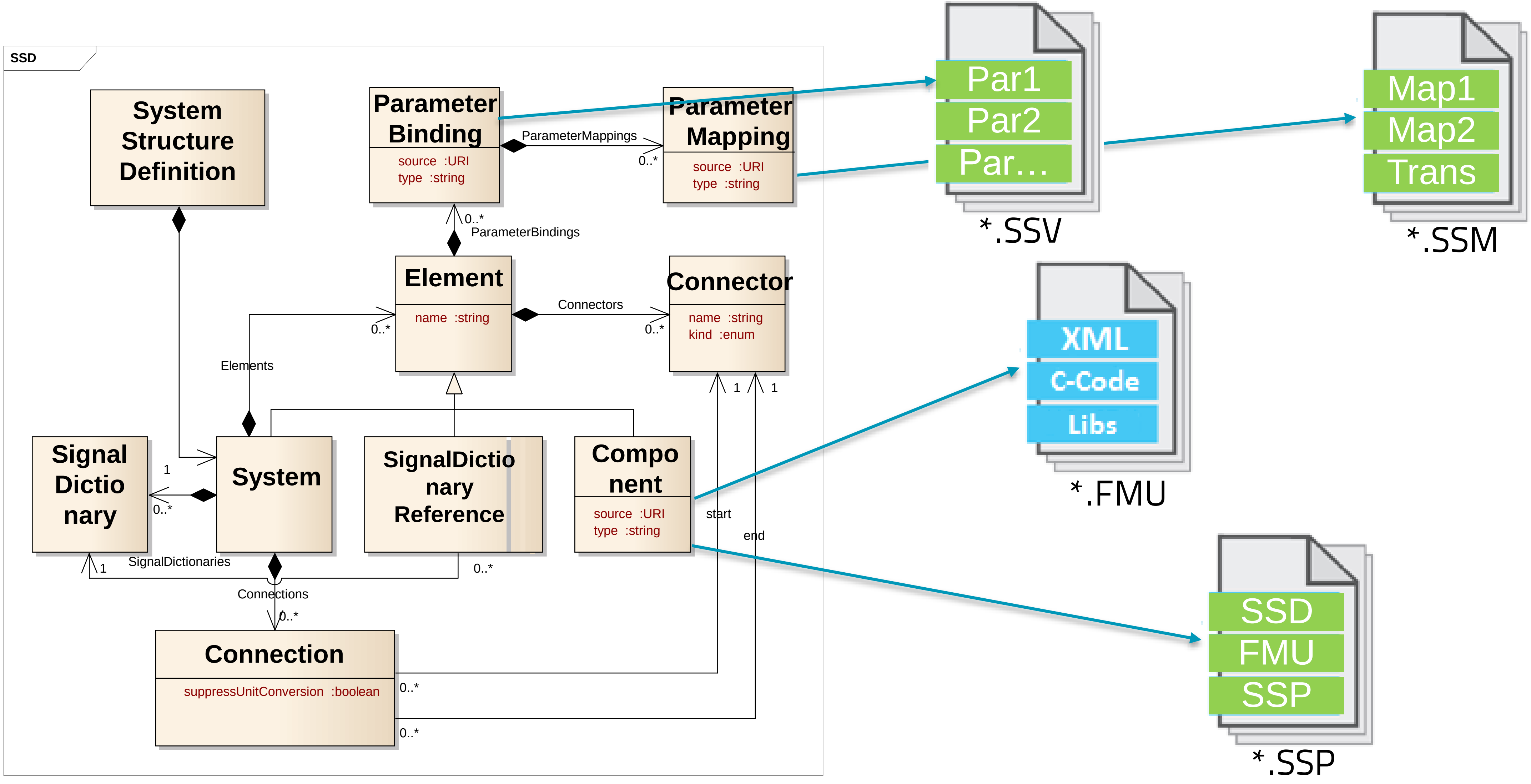
- Collected on a meeting with BMW, Bosch, ZF, PMSF (2014):
  - No possibility to separate parameter data from the FMUs
  - No possibility to change parameters in a consistent way independently from the integration environment for single FMUs.
  - No possibility to handle intellectual property of parameters
  - No possibility of mapping parameters in a network of FMUS
  - No possibility to store a network of FMUs tool independently

Main Purposes of SSP

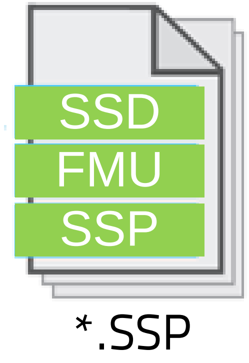
- Define a standardized format for the **connection structure of a network of components**.
- Define a to these components**standardized way to store and apply parameters** .
- The developed standard / APIs should be usable in all stages of development process (architecture definition, integration, simulation, test in MiL, SiL, HiL).
- The work in this project shall be coordinated with other standards and organizations (FMI, ASAM, OMG).

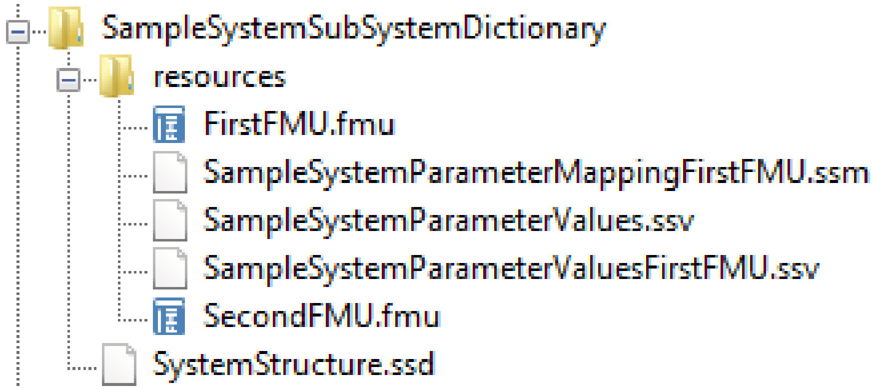


Overview of XML Schema Definitions



XML Schema Description - System Structure Package





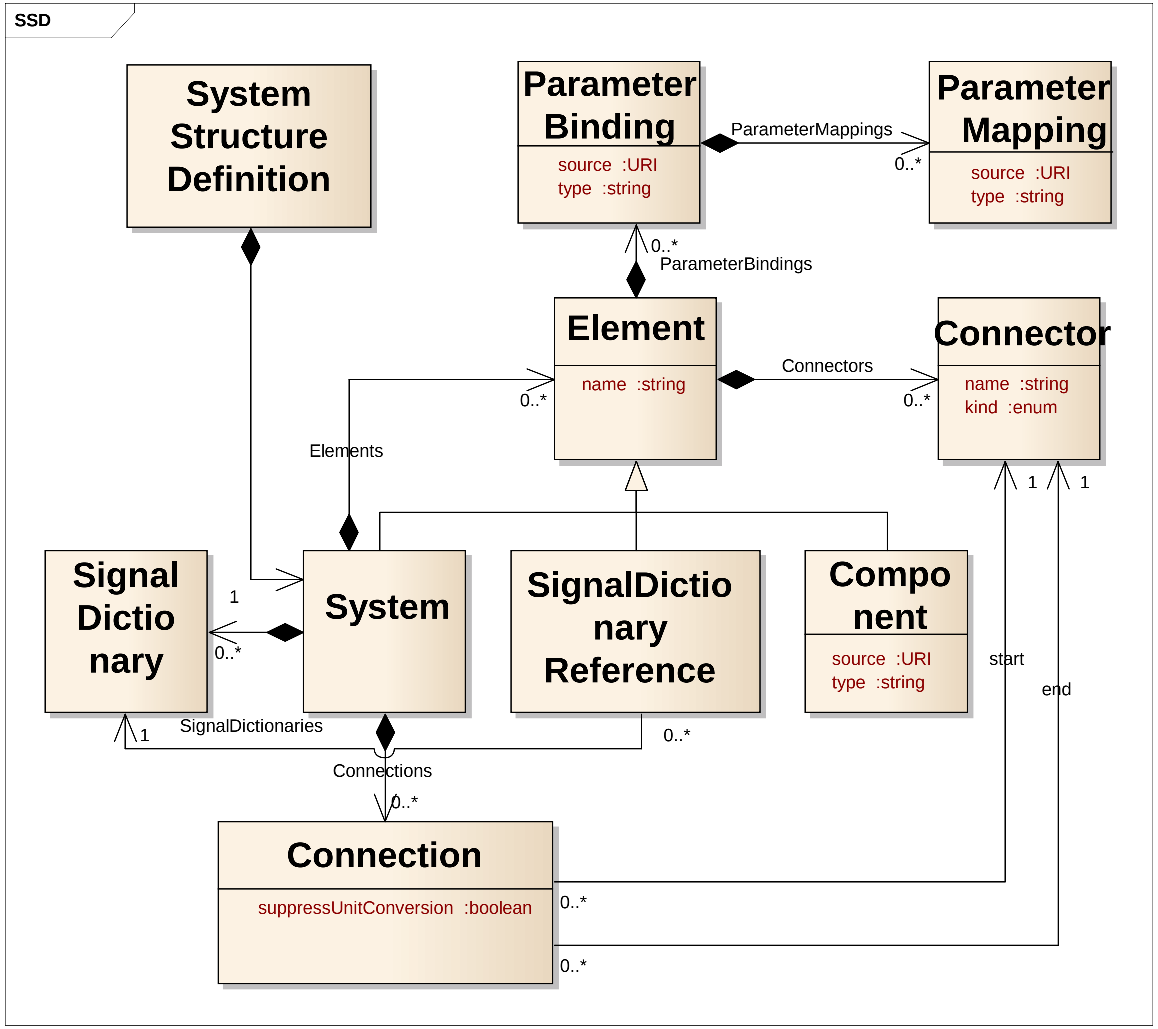
Use case

- Exchange of Complete Systems with Variants

Features

- All information (FMUs, system structure definition, parameters) can be stored in one archive (zip-file)
- Multiple SSDs in one SSP allows for variant modeling

XML Schema Description - System Structure Definition



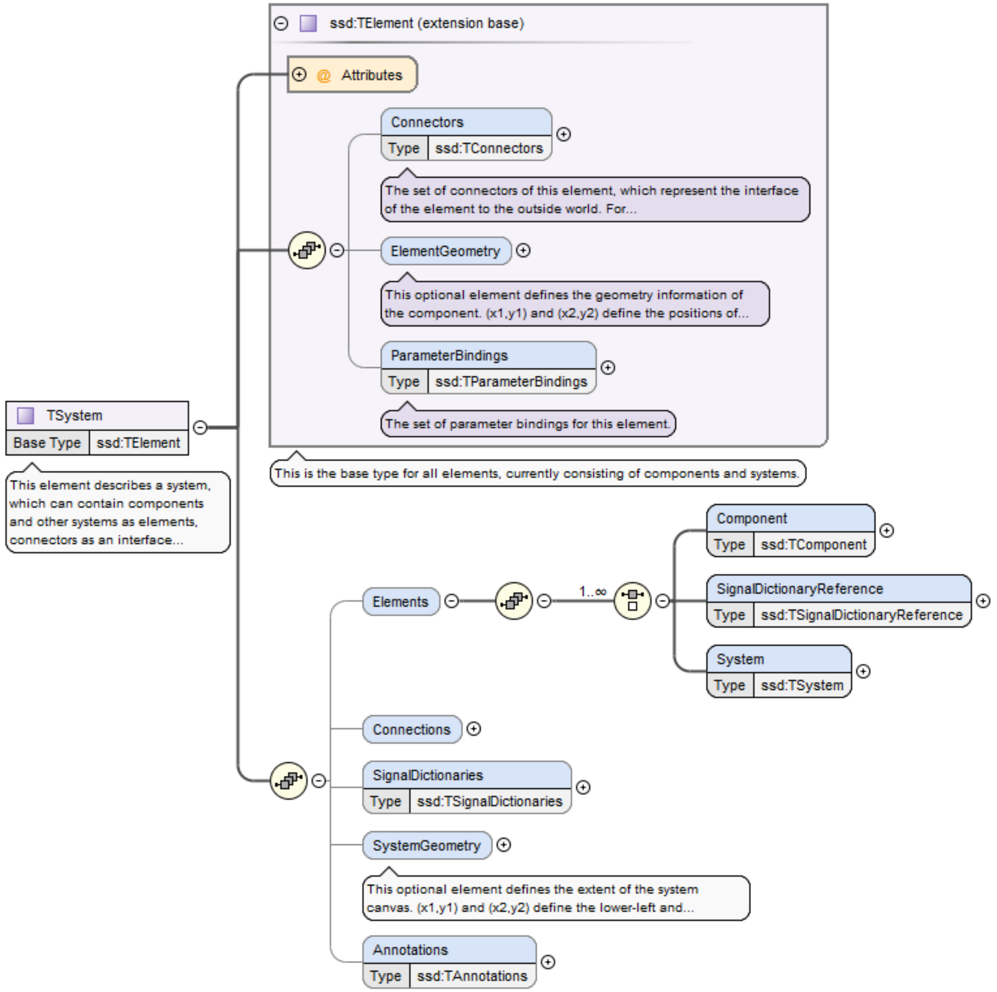
Use case

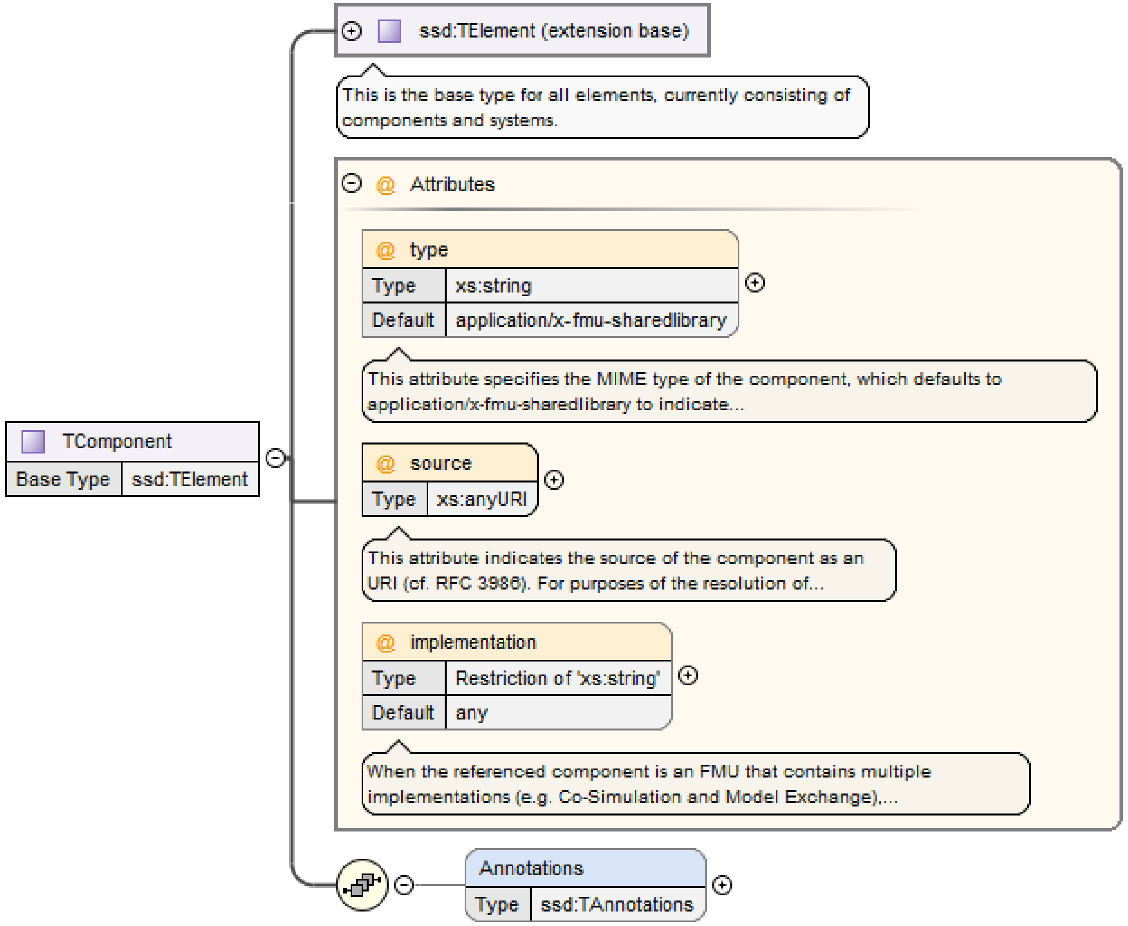
- Defining a Network of FMUs

Features

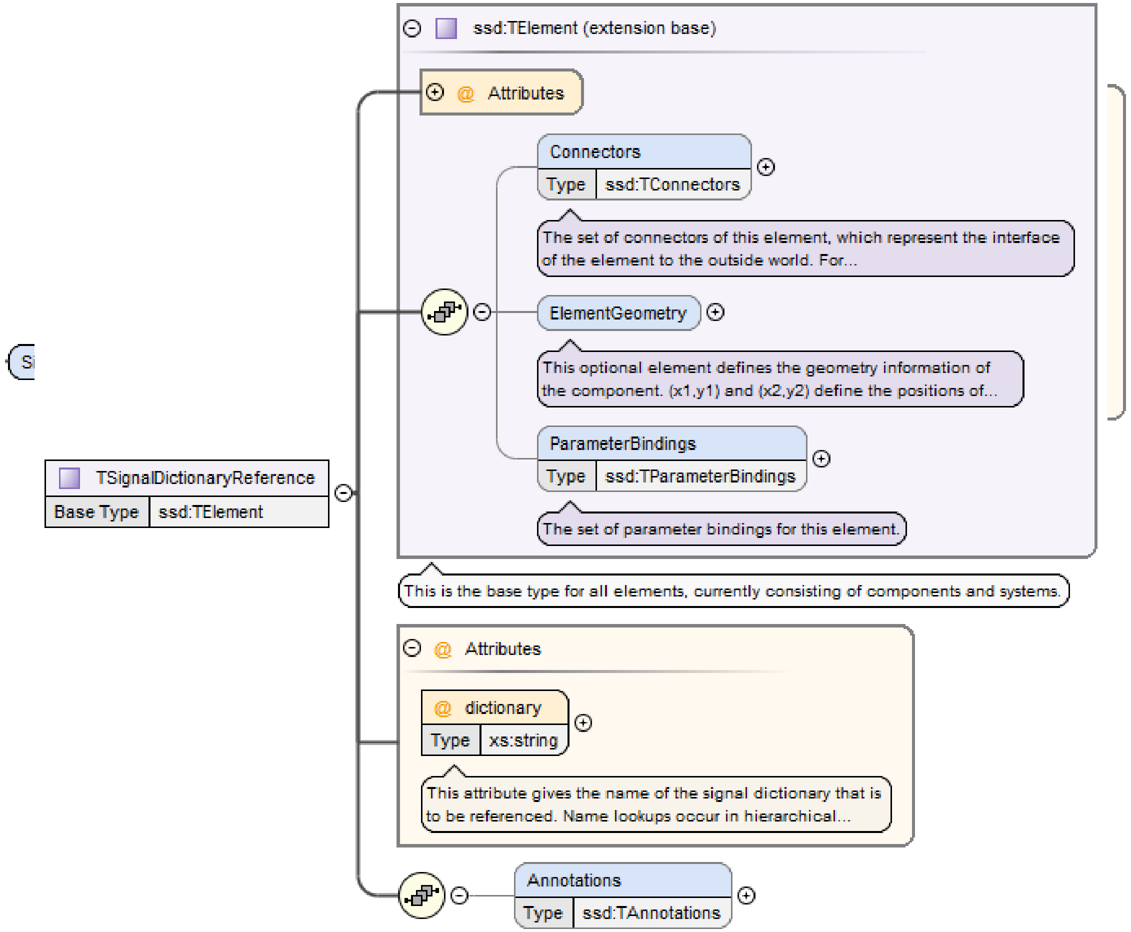
- Hierarchical sub-systems
- Empty components/FMUs as interface templates
- External resources via URIs: Both relative to SSD/SSP or absolute, e.g. via HTTP(S).
- Connections with unit conversions and optional linear/map transformations
- Optional: Diagram geometry

XML Schema Description - System Structure Definition





XML Schema Description - Signal dictionaries



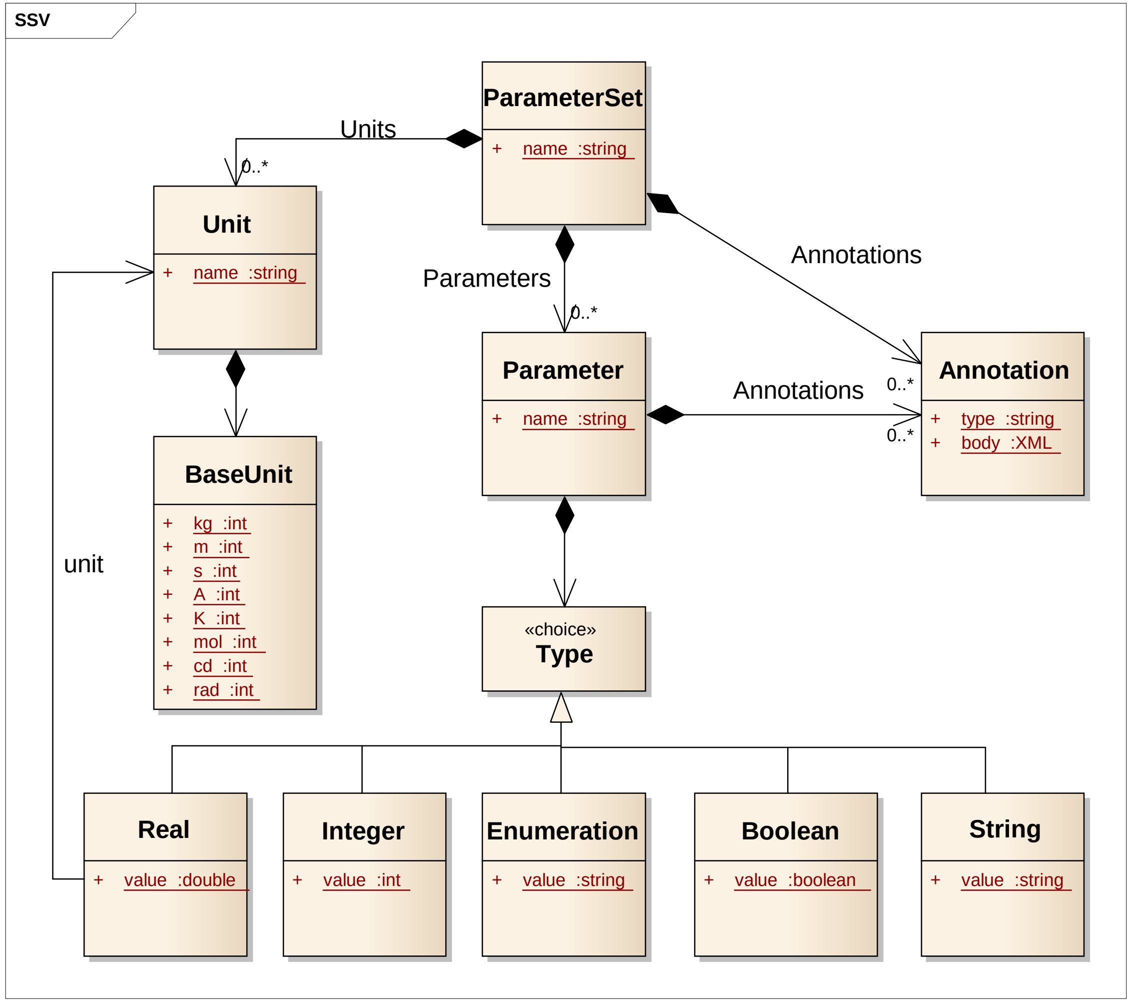
Use cases

- Collecting Control Signals in a Central Location

Features

- Causality is checked by tool automatically
- Crosses hierarchies without need for downward passing
- Well-suited for e.g. ECU control busses

XML Schema Description – Parameter Values Data



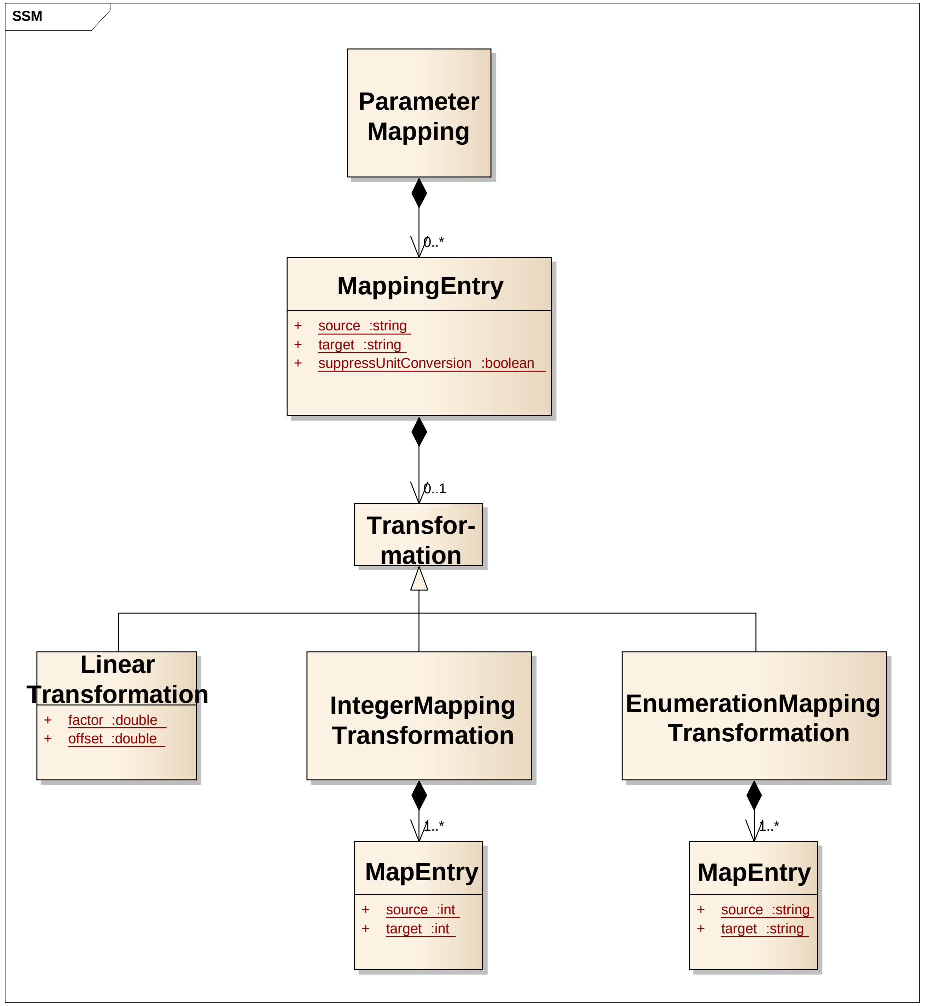
Use case

- Tool-independent Exchange of Parameter Data

Features

- Neutral exchange format between parameter sources
- Compatible to FMI standard
- Provides some meta data
- Access to param DBs via HTTP (-> Parameter API)

XML Schema Description - Parameter Mapping



Use case

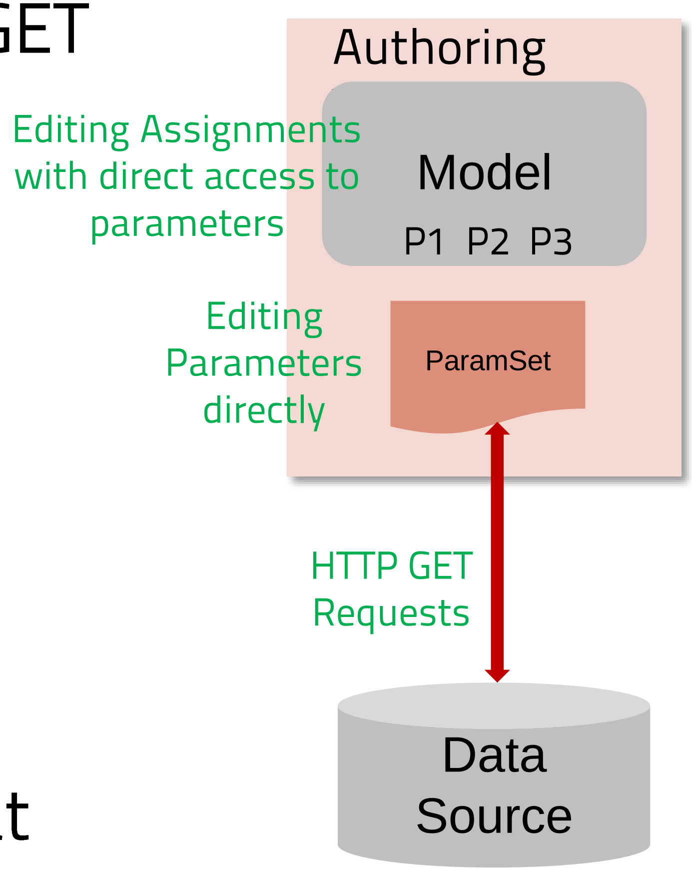
- Mapping Parameters to FMUs when the Parameter Names differ or Parameter Values require Transformations

Features

- Can be stored separately from System Structure and Parameter Data
- Can be inlined into SSD
- Optional manual linear and mapping transformations

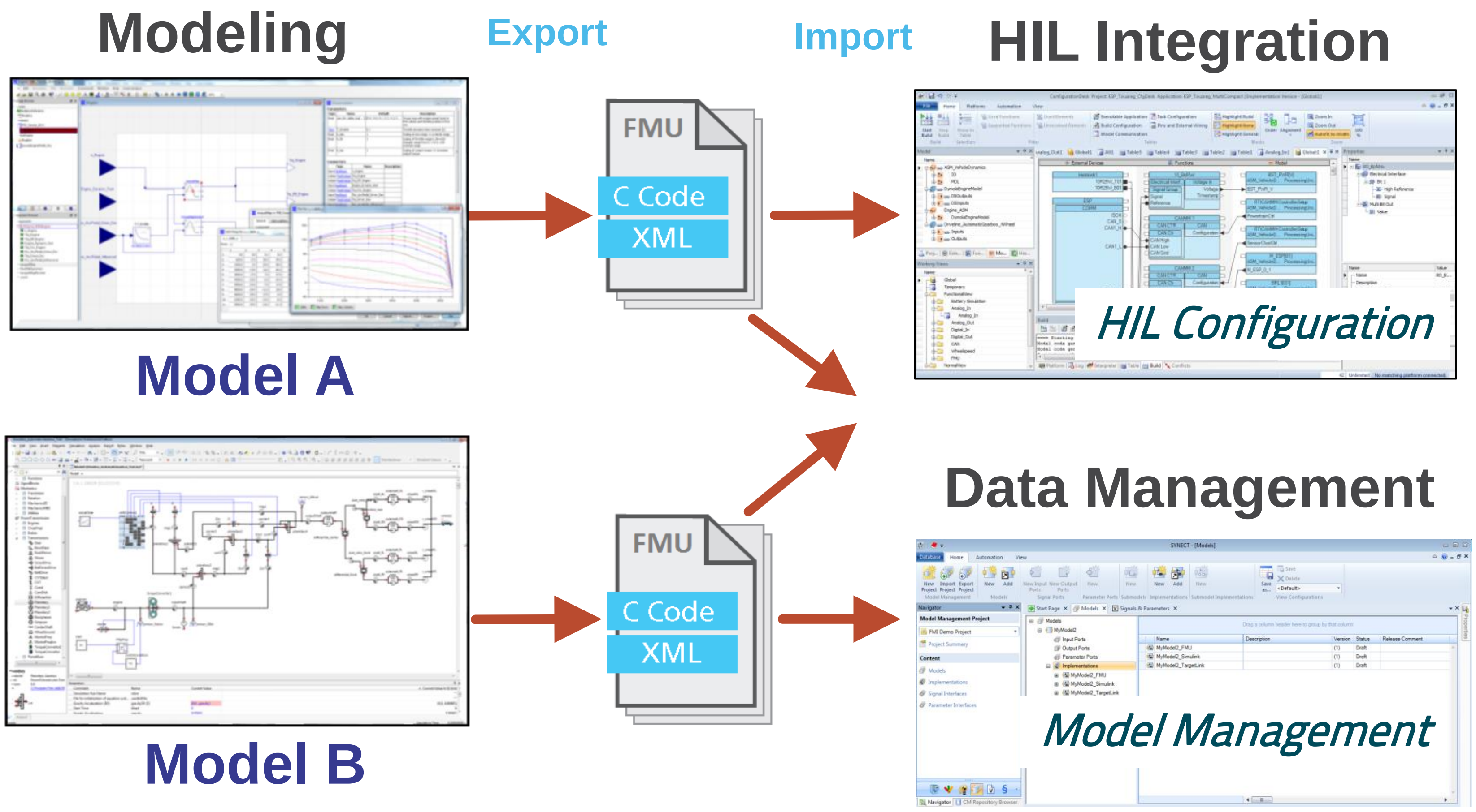
Parameter API Get Mechanisms

- General Idea:
  - Access to external parameter sources via HTTP(S) GET Requests
  - Request URI is the source attribute
  - Type attribute passed via accept request header
  - Updates handled efficiently via ETag/Conditional GET/HEAD
- Returns Parameter Data in the format requested:
  - application/x-ssp-parameter-set -> SSV file format
  - Sources and tools can support other formats



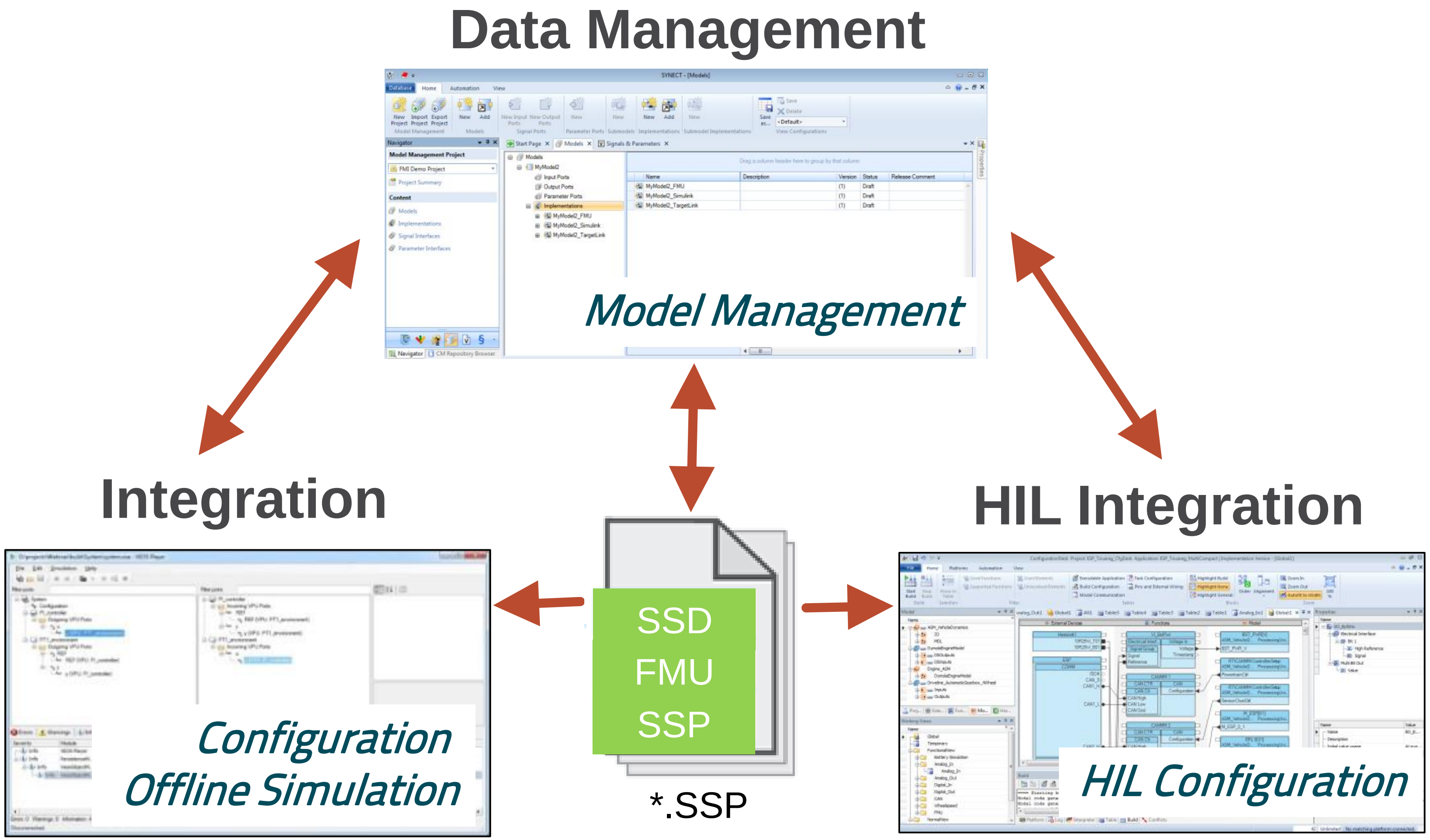
Integration of FMUs for HIL Testing

- HIL configuration tools are importing FMUs to integrate them with other FMUs, Simulink-based models and real ECUs
- Data Management tools are managing the lifecycle of the FMUs



Reuse of the System Structure for SIL, MIL and HIL

- Integration and Data Management tools share a vendor independent system description (SSP)
- Reuse of tools, configurations, models, tests, layouts and parameters at system level is supported

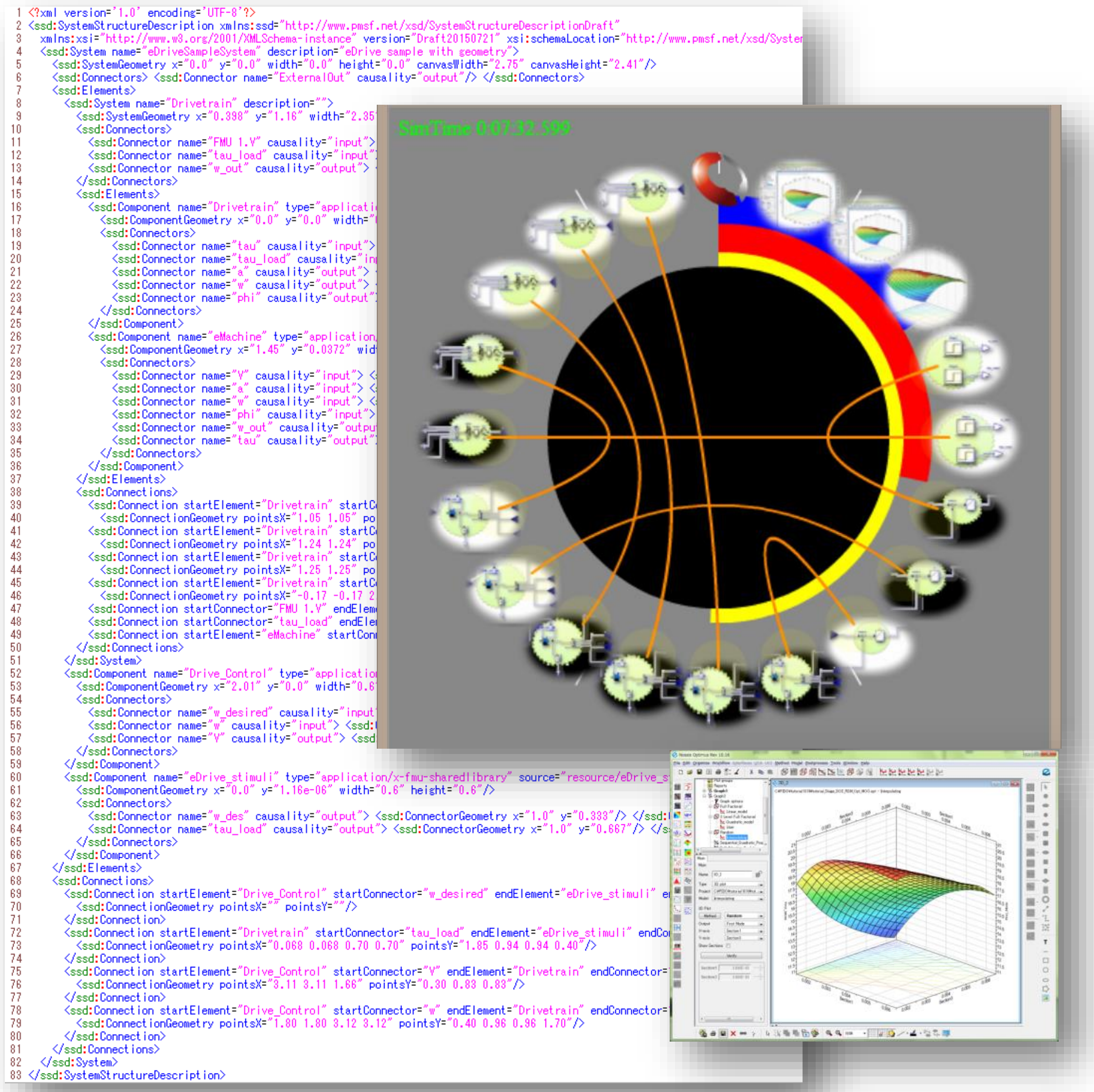


Commercial Prototypes

- Model.CONNECT™ by AVL: model integration and execution
- FMI Integrator by Modelon: aggregating FMUs to networks of "Super-FMUs"
- FMI Bench by PMSF: Workbench for FMUs

Prototypes – Online Testing Tool

- Scalablility of <ssd:Connectors>
  - Ring configuration at a glance
  - 3D Flash UI for <ssd:Component>
- Time integration control master
  - Unit Test with default parameter
  - Synchronized Co-Simulation Test
- Parameter database as FMU
  - FMU of (sqlite.DB + sql.DLL)
  - exported by Optimus®



Future work / Outline

- Further Development of API for parameter handling
- Try to involve providers of simulation data management systems in this project
- Evaluate approaches with „real-world examples“
- Publish first release soon

- Any contribution is very appreciated !