FMI, Layered Standards and ASAM Standards - Enabling Seamless SiL Simulation of Virtual ECU

International ASAM Conference 2024, Dec 4th /5th 2024

Christian Bertsch (Bosch)
Kahramon Jumayev (Akkodis)
Pierre Mai (PMSF)
Benedikt Menne (dSPACE)
Masoud Najafi (Altair)
Tim Pfitzer (Bosch)
Jan Ribbe (Synopsys)
Klaus Schuch (AVL)

Overview

- The Functional Mock-up Interface 3.0

- Layered Standard concept

Three Layered Standards by the FMI Project

- Layered Standard for XCP

- Layered Standard for Network Communication

- Layered Standard for Structured Data

- Demo from different SiL tool vendors illustrates cross-manufacturer interoperability of FMI 3.0 and these layered standards.

The Functional Mock-Up Interface

The Functional Mock-up Interface is a free standard that defines a **container** and an **interface**

- to exchange **dynamic simulation models**
- using a combination of **XML files, binaries** and **C code**, distributed as a **ZIP file.**
- Current releases: **FMI 2.0.5** and **FMI 3.0.2**
- **200+ tools** and libraries support FMI
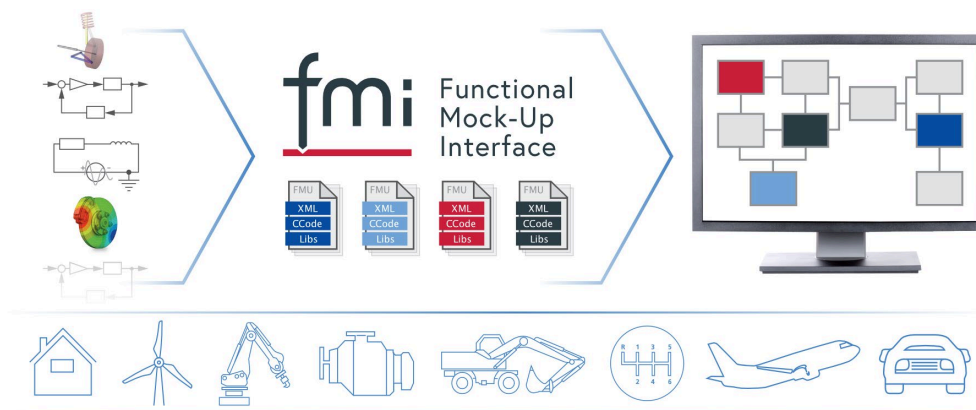
FMI Project

within the Modelica Association

Motivation for FMI 3.0:

New use-cases and requests:

- Virtual Electronic Control Units (**vECUs**):
  - better support needed!
- **Advanced Co-Simulation**
  - improved co-simulation methods are needed to improve performance and accuracy
- Multi-FMU simulations are getting more common
  - **Events** necessary in complex control systems
  - Events must be synchronized across FMUs
- New ML, **AI, optimization** applications
  - More derivatives computations is required

Main Improvements:

Performance Accuracy
- Event mode for Co-Simulation
- Intermediate variable update
- Clocks
- New variable types
- Array variables

New Applications
- Terminals and icon
- FMI for Scheduled Execution (SE)
- Adjoint derivatives
- **Preparation for Layered Standards**

Layered Standards

- Layered standards **extend the FMI standard** for **new application domains**..
- They can be defined by different organizations

Layered Standards in development by the FMI Project:

- **FMI-LS-XCP:** for XCP support

- **FMI-LS-STRUCT**: for structuring of variables + lookup tables

- **FMI-LS-BUS:** for network communication

Demo!

- **FMI-LS-REF:** description of basic validation experiments and other files attached to an FMU

FMI-LS-XCP: Layered Standard for XCP support

XCP (Universal Measurement and Calibration Protocol) is a network protocol originating from **ASAM** for connecting calibration systems to electronic control units, ECU

**Version v1.0.0 released on Dec. 3rd 2024!**

Main idea:

- ship an **A2L file** according to the ASAM ASAP2 in standardized location inside the FMU

- **describe the capabilities** w.r.t. the XCP protocol

ECU

SiL Simulation

vECU   *Plant model*

XCP Calibration Tool
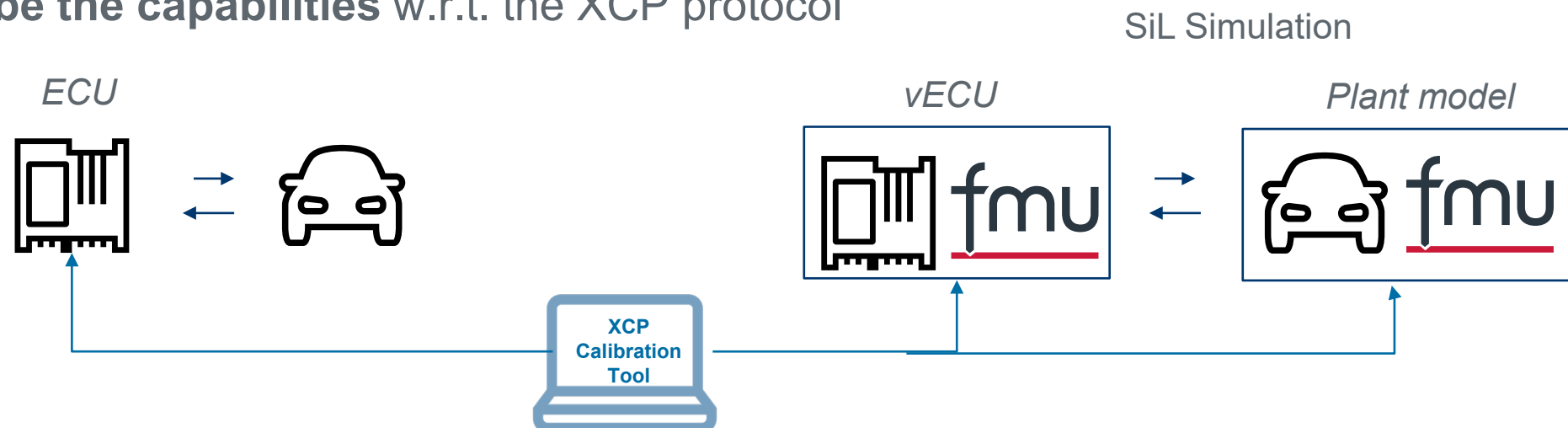
FMI LS-STRUCT: Layered Standard for Variable Structuring

Current scope:

- **Maps sampled on rectilinear grids**
  using FMI 3.0 arrays by adding meta-data to define

  - Domains

  - Codomain

  - Optionally: Variables referencing current operating point, …

  - Support for Modelica Standard Library CombiTables

  - Compatible with ◈ **ASAM**-MCD2 MAPs, CURVEs, …

- Maps sampled on an irregular grid ("Point Cloud")


- Using "FMI 3.0 Terminals" for grouping

Codomain **z**

Input
in_y   Domain **y**

Current
opererating
point

Input
in_x   Domain **x**
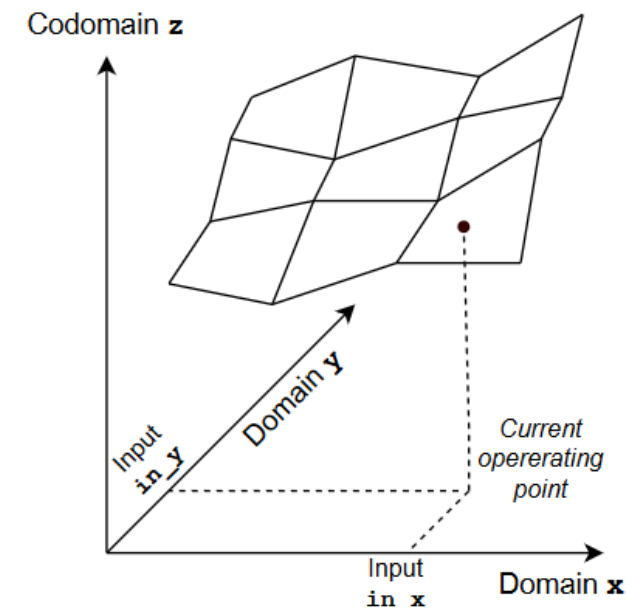
# FMI-LS-BUS: Layered Standard Network Communication

Simulation of automotive systems requires **network communication between vECUs.**

Idea:

- **Use FMI 3.0 core standard features** to specify a common bus interface, using **Co-Simulation**, **Clocks** and **Terminals**

- Support **automotive networks protocols** and others from **different industries** (CAN, LIN, FlexRay, Ethernet, …)

- Using network description formats DBC, LDF, ◈ **ASAM** FIBEX, ARXML

- **Two variants:**

  - **Physical abstraction layer ("high cut"):** Simply transport physical signal values between virtual ECUs by using unit-based and clocked variables

  - **Network abstraction ("low cut"):** Detailed emulation of a specified bus system to realize virtualized bus driver implementations, including feedback from the physical drivers about transmission status or network node states
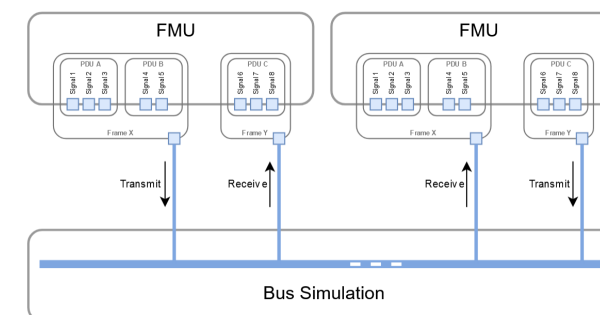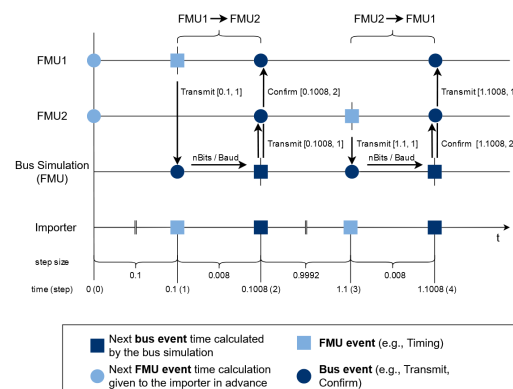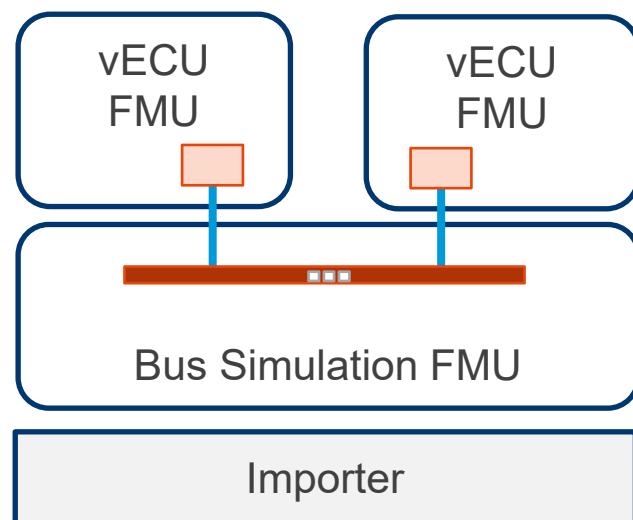
FMI-LS-BUS: Layered Standard Network Communication
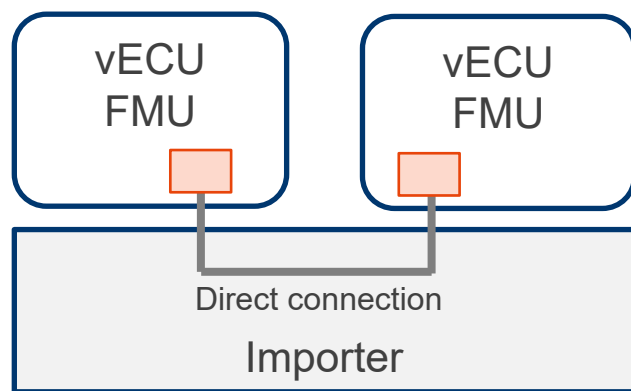
Bus simulation idealized

- 1:1 connection
- Idealized bus
  (only ideal timing, no bandwidth limits, …)
- Standard FMI Importer can be used

Bus simulation via Bus Simulation FMU

- n:m connection
- Bus behavior can be simulated
- Standard FMI Importer can be used
- Can be generated based on network descriptions
- Changes of on system level require regeneration
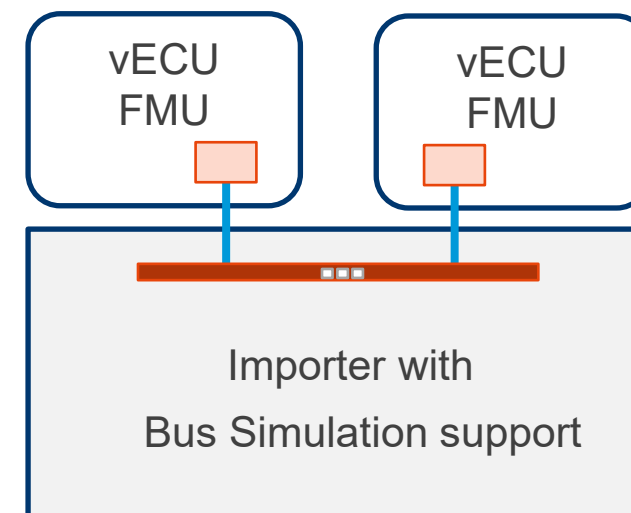
Bus simulation supported by the Importer

- n:m connections
- Bus behavior can be simulated
- Dynamic configuration
- No network descriptions needed

FMI-LS-BUS: Layered Standard Network Communication

FlexRay

Specification and verification of FlexRay support for Network Abstraction

LIN

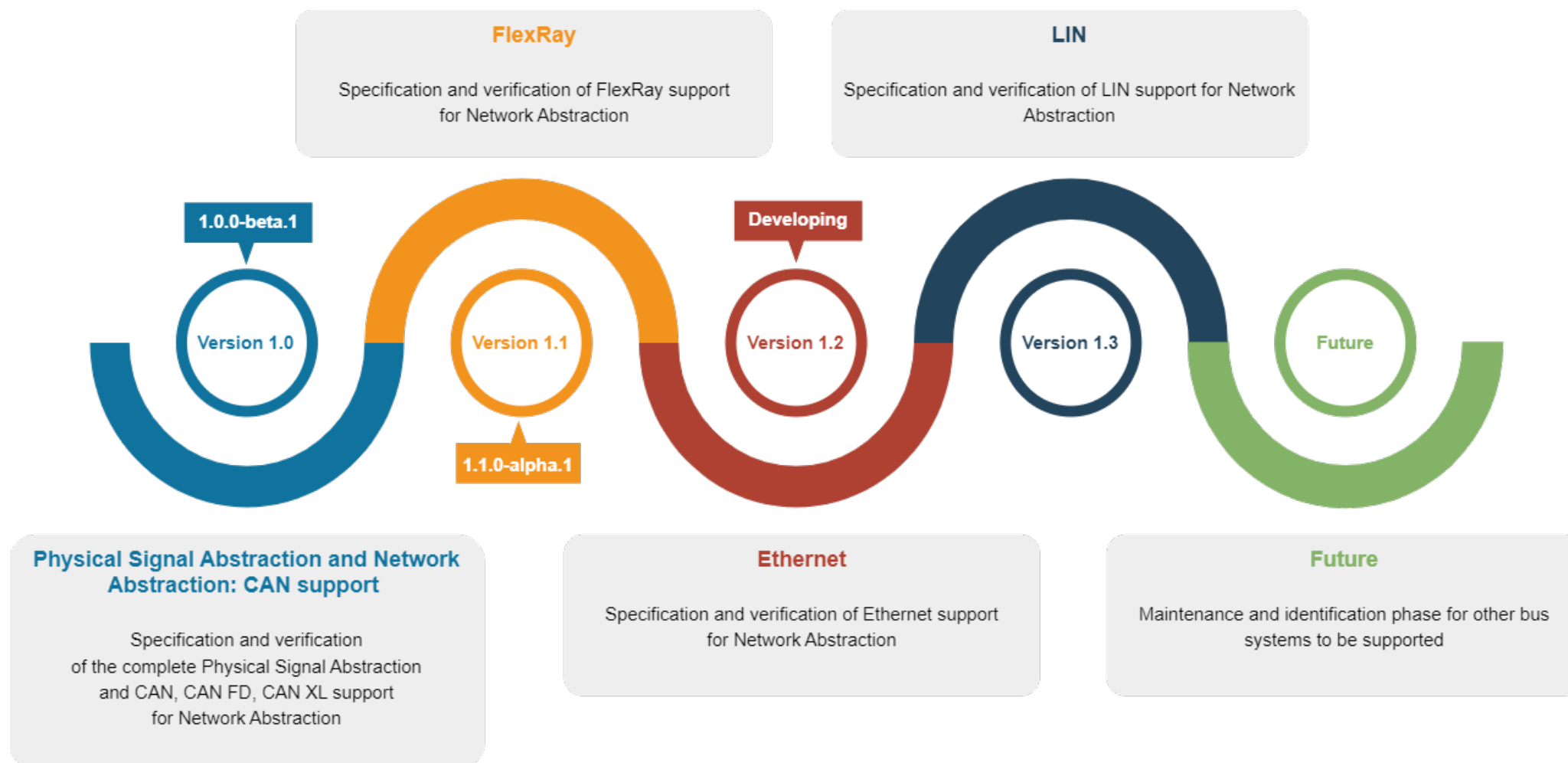Specification and verification of LIN support for Network Abstraction

1.0.0-beta.1

Developing

Version 1.0

Version 1.1

Version 1.2

Version 1.3

Future

1.1.0-alpha.1

Physical Signal Abstraction and Network Abstraction: CAN support

Specification and verification of the complete Physical Signal Abstraction and CAN, CAN FD, CAN XL support for Network Abstraction

Ethernet

Specification and verification of Ethernet support for Network Abstraction

Future

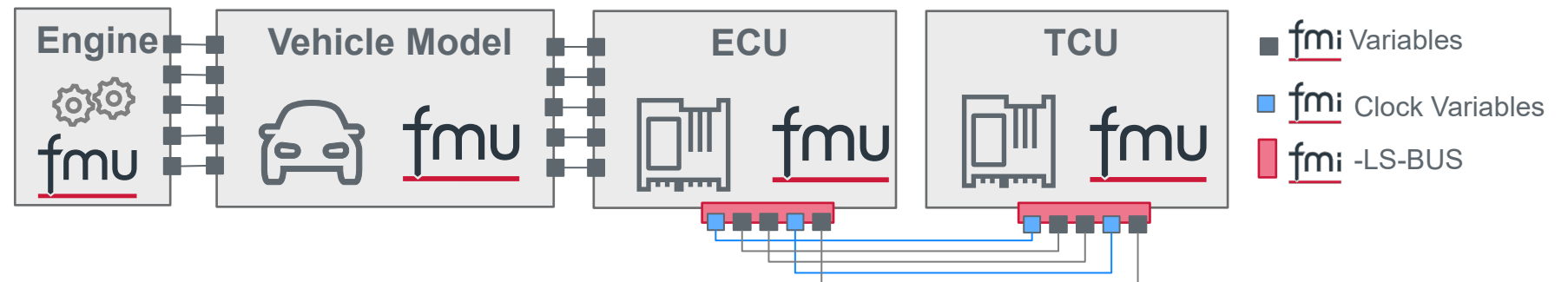Maintenance and identification phase for other bus systems to be supported

FMI-LS-BUS: High Cut Demonstrator

Closed-loop fuel injection control system

Components:

- **TCU:** Transmission Control Unit

- **ECU:** Engine Control Unit – Here only used as a bus adapter

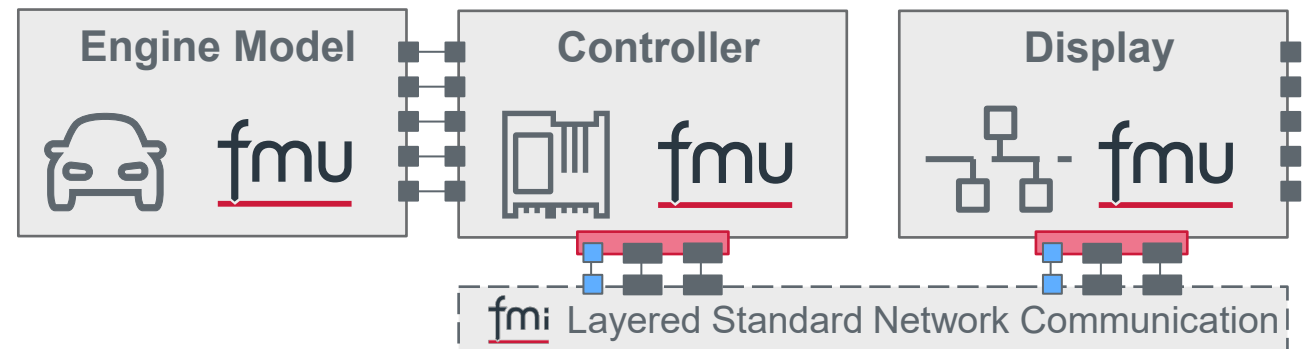- **Vehicle Model incl. Engine Model:** Simplified vehicle model

FMI-LS-BUS: Low Cut Demonstrator

Closed-loop fuel injection control system

Components:

- **Controller:** AUTOSAR based virtual electronic control unit (vECU-Type3)
  including BSW configuration and components e.g. PDUR, CAN-IF & FMI3lsBus Driver

- **Engine Model:** Simplified vehicle model
  including relevant sensor and actuator functions

- **Display:** Restbus simulation model
  including simplified PDU scheduler
  (non-AUTOSAR) and signal to bus interface
  & FMI3lsBus Driver

Demonstration:

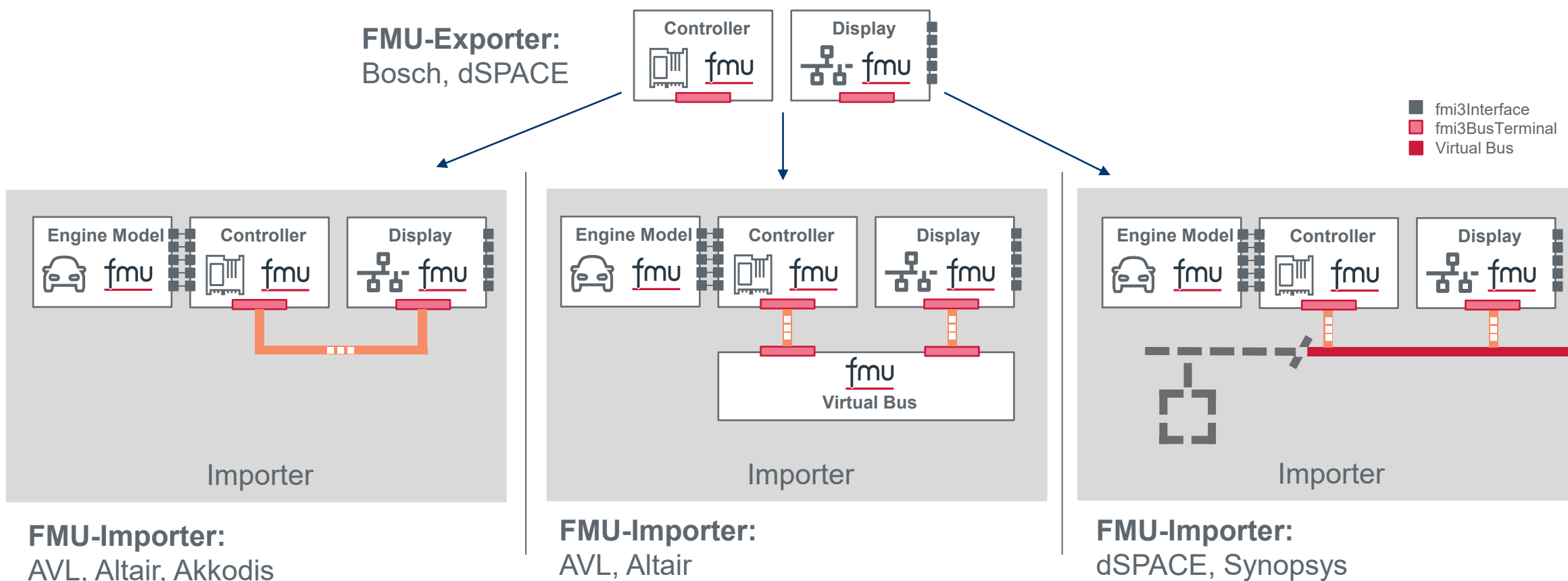- **Seamless Tool Interoperability:** Demonstrating FMUs from diverse tools and vendors integrated into various simulation environment

- **Configurable Virtual Bus Setup:** Showcasing how the bus setup can adapt to various contexts without the need for re-exporting FMUs

■ fmi Variables
■ fmi Clock Variables
■ fmi -LS-BUS

Engine Model — **Controller** — **Display**

fmi Layered Standard Network Communication

FMI-LS-BUS: Low Cut Demonstrator

Closed-loop fuel injection control system

FMU-Exporter:
Bosch, dSPACE

fmi3Interface
fmi3BusTerminal
Virtual Bus

Virtual Bus

Importer

Importer

Importer

FMU-Importer:
AVL, Altair, Akkodis

FMU-Importer:
AVL, Altair

FMU-Importer:
dSPACE, Synopsys

* reduced XCP focused setup without bus simulation

Take-Away: (R)evolution to SiL Simulation of Virtual ECUs

- The FMI standard alone is available for exchanging simple virtual ECUs in early development phases.

- For more complex vECUs this standard is not sufficient, especially if these includes **Bus Interfaces** and **XCP** requirements.

- To close this functional gap the FMI project and partners are **working intensively** on the **FMI-LS-BUS**, **FMI-LS-XCP** and **FMI-LS-STRUCT** that enables efficient **exchange** between partners for such **virtual ECUs**.

- The advantages for validation are obvious:
  - **Lower integration costs** thanks to **better interoperability** of tools and test platforms from **different vendors** within virtual validation scenarios
  - More **efficient collaboration** thanks to **significantly reduced coordination effort** between **OEMs and suppliers**

Questions?

Comments?

Ideas?

Feedback?

Join the FMI LinkedIn Group!