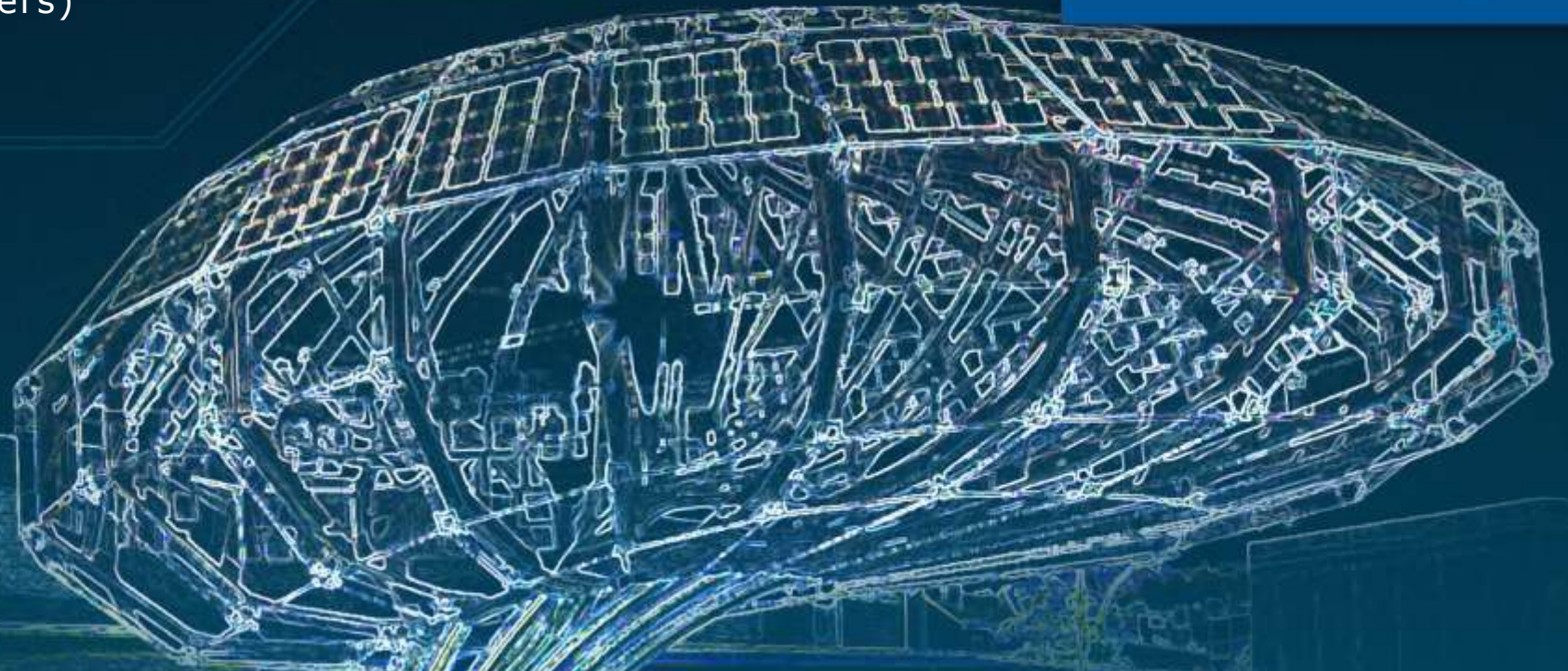


AVL



AVL List GmbH (Headquarters)



# AVL @ OpenADx

Joining the Eclipse Working Group

**Zehetner, Balic,  
Krasser**

# Our motivation for OpenADx

## Historical

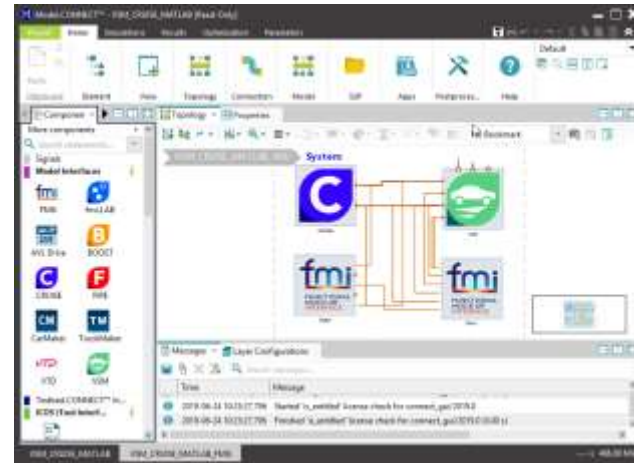
- More than 30 years of simulation on testbeds, more than 20 years RT/non-RT Co-Simulation
- Co-simulation
  - Research started at VIF (ICOS) in 2004, ICOS V1.0 released in 2010
  - Integration of ICOS in Model.CONNECT™ since 2014
  - First release of Model.CONNECT™ in 2015, current release 2019 R1
  - Support of FMI since 2012, SSP since 2018, DCP prototype (project ACOSAR)
  - Integration of ~30 simulation tools (tools wrappers)
- Real-time Co-simulation
  - Coupling of real-time and soft real-time, research projects ACORTA [1..3], AVL, Porsche, VW, ViF, started in 2012
  - As part of Testbed.CONNECT™ since 2016

**But: huge effort for handling wrappers for new tools, different versions**

→ **OpenADx**

# Model.CONNECT™ architecture

Model.CONNECT™ currently uses to co-simulation engines in the background:



## MCX

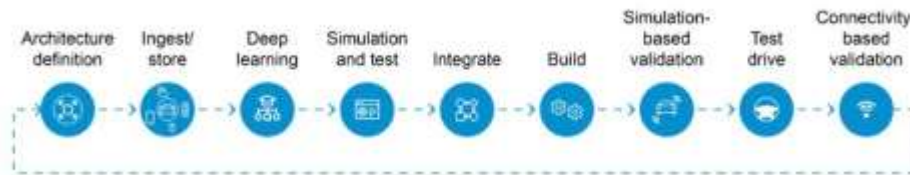
- Owner is AVL
- Couples component models within one process → intra-process co-simulation
- Support of FMU and other standards (SSP, DCP prototype etc.), AVL tools and some other commercial tools.

## ICOS

- Owner is VIF
- Couples component models by coupling of tools → multi-process co-simulation
- Support of MCX, broad range of 3<sup>rd</sup> party simulation tools, real-time co-sim

# Our next steps towards OpenADx...

- We think, OpenADx is a great idea!
- We want to contribute to OpenADx
- We would be highly interested to coordinate the “Integrate” topic within OpenADx (and we think we have the knowledge for it)



- By joining OpenADx we want to push AVL’s open source involvement to a new level, therefore...

...we are happy to announce...

...that AVL plans to open source the co-simulation engine MCX within 2019...

...supporting intra-process co-simulation of FMUs, open standards, generic user-code, etc....

# Open source co-simulation platform within OpenADx



AVL's open source strategy for MCX (first steps until end of 2019)

- MCX co-simulation engine
- Support of FMI for Co-Simulation & Model Exchange
- Generic numerical input blocks
- Standards interface support (DCP, ROS2 prototypes etc.)
- Environment Simulation interface (e.g. SUMO, VTD or similar)
- Generic user code block for easy wrapper interface development

Further topics we see as relevant (after 2019): multi process co-simulation, eFMI, ...

*But of course, this should be driven by the community...*

More details to come at our release event in Q3/4 2019

Thank You



[www.avl.com](http://www.avl.com)

