Powertrain System Simulation

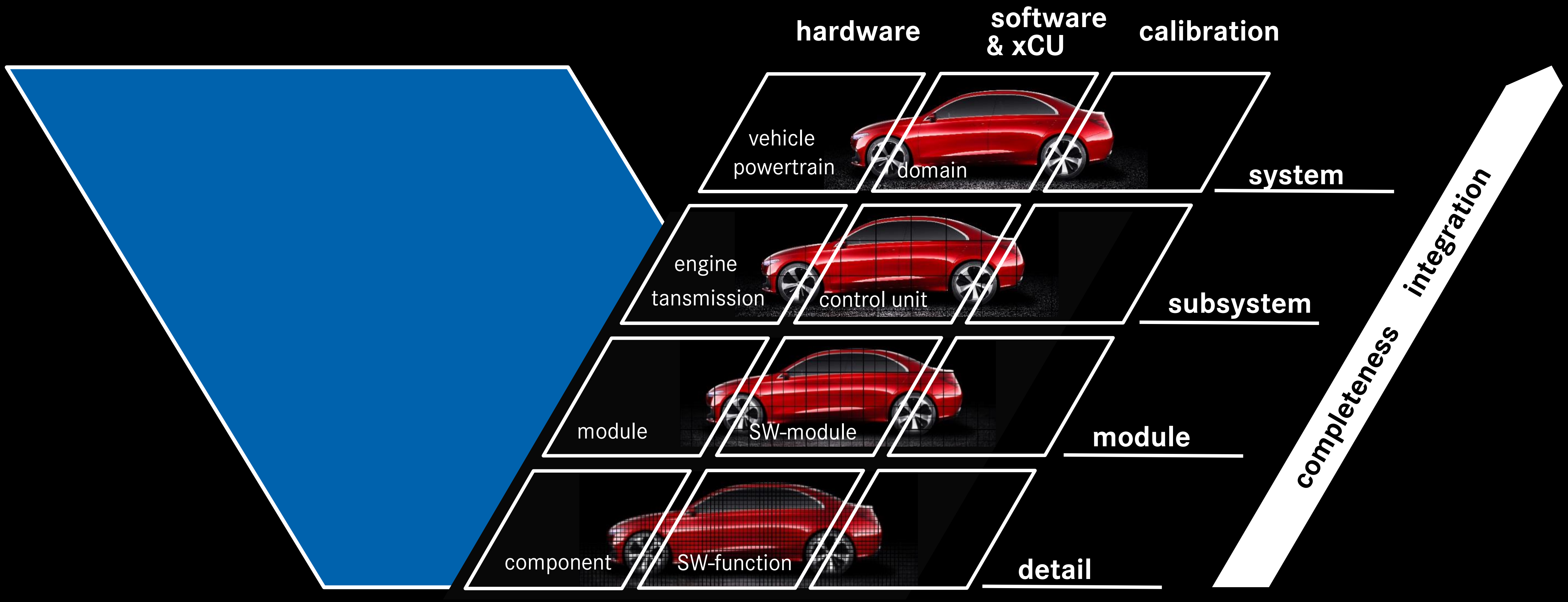
Frank Uphaus, Renè Linssen | Daimler AG
QTronic User Conference 2018 - Virtual ECUs and Applications
October 18th, 2018, Berlin.

Mercedes-Benz
Das Beste oder nichts.

Powertrain System Simulation

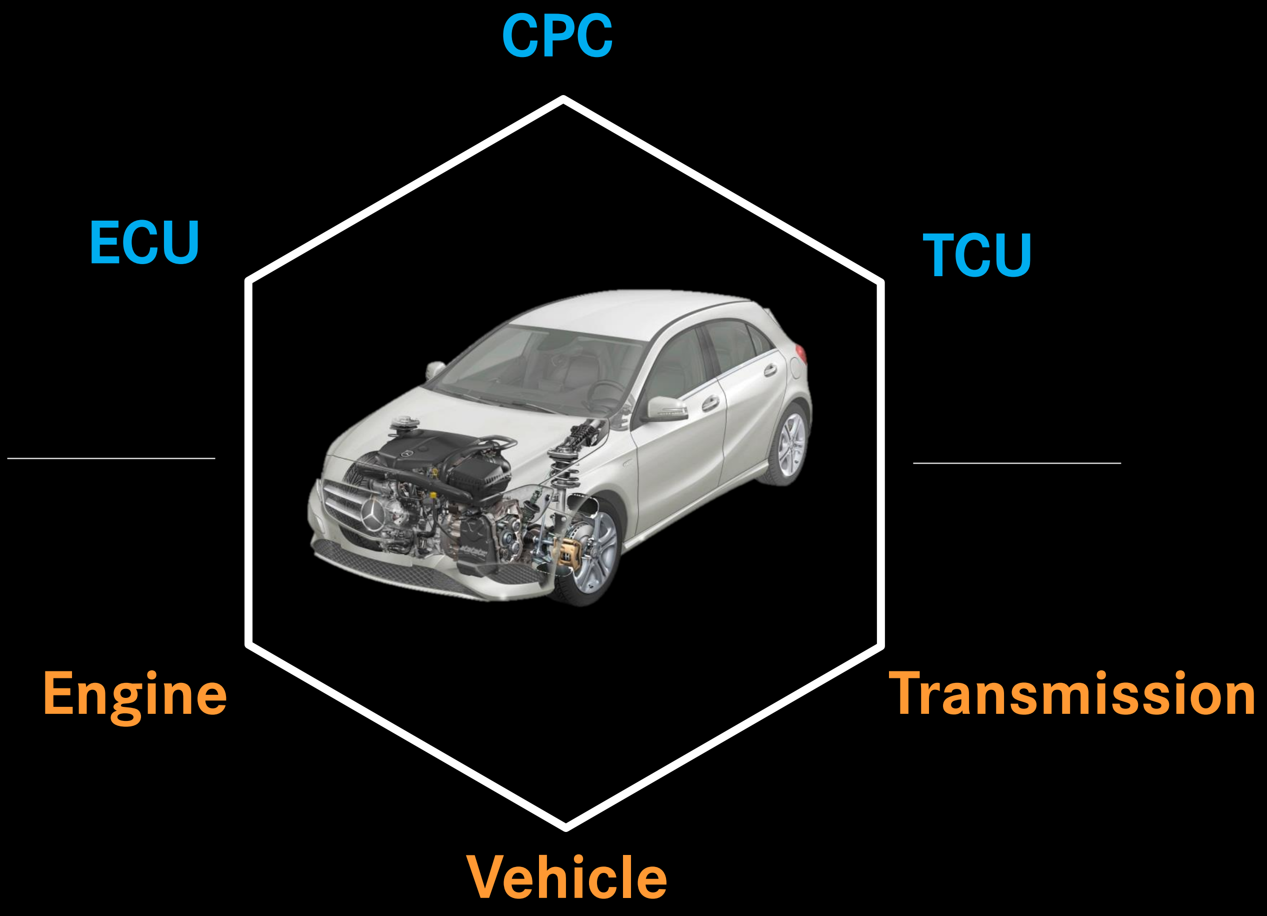
1. Vision
2. Concept
3. Implementation
   - Virtual Control Modules
   - Plant Models
   - Use Cases & Testing
4. Conclusions on Business Model
5. Summary

Integration and Completeness



Aim for completeness of the virtual approach for maximum impact!

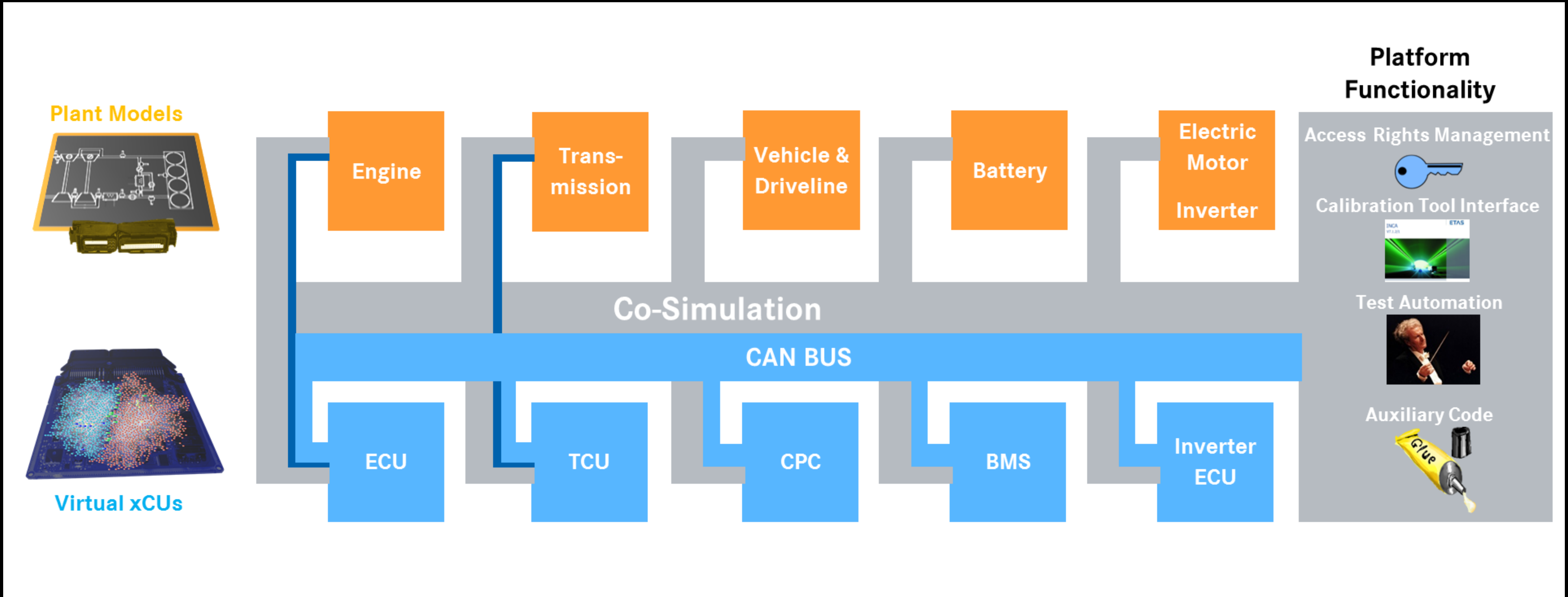
Vision



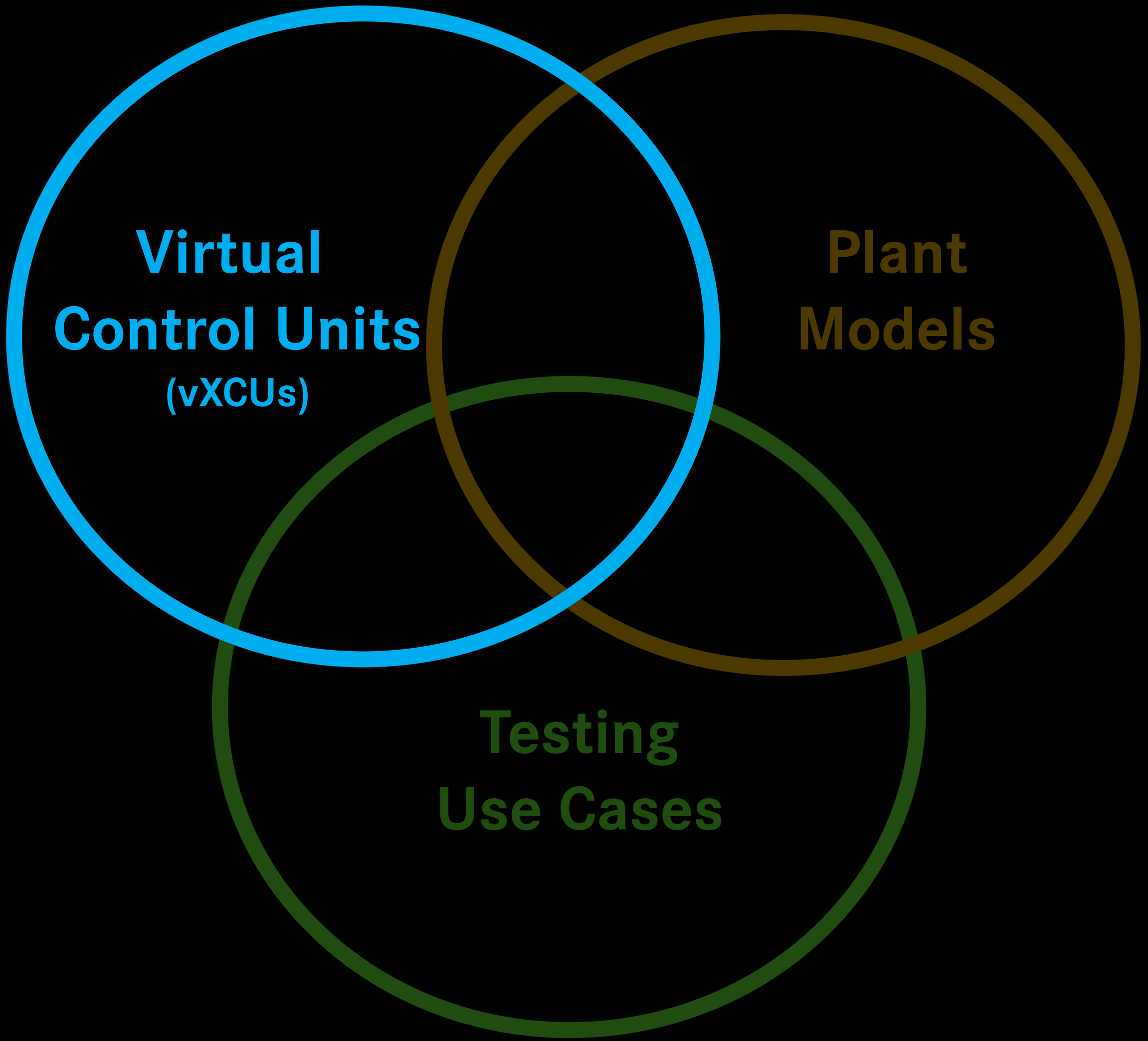
* CPC = Central Powertrain Controller

To calibrate drivability (also) using a virtual vehicle the whole powertrain including all functional software and hardware components has to be digitalized.

SiL Platform for Powertrain System Simulation



Ingredients

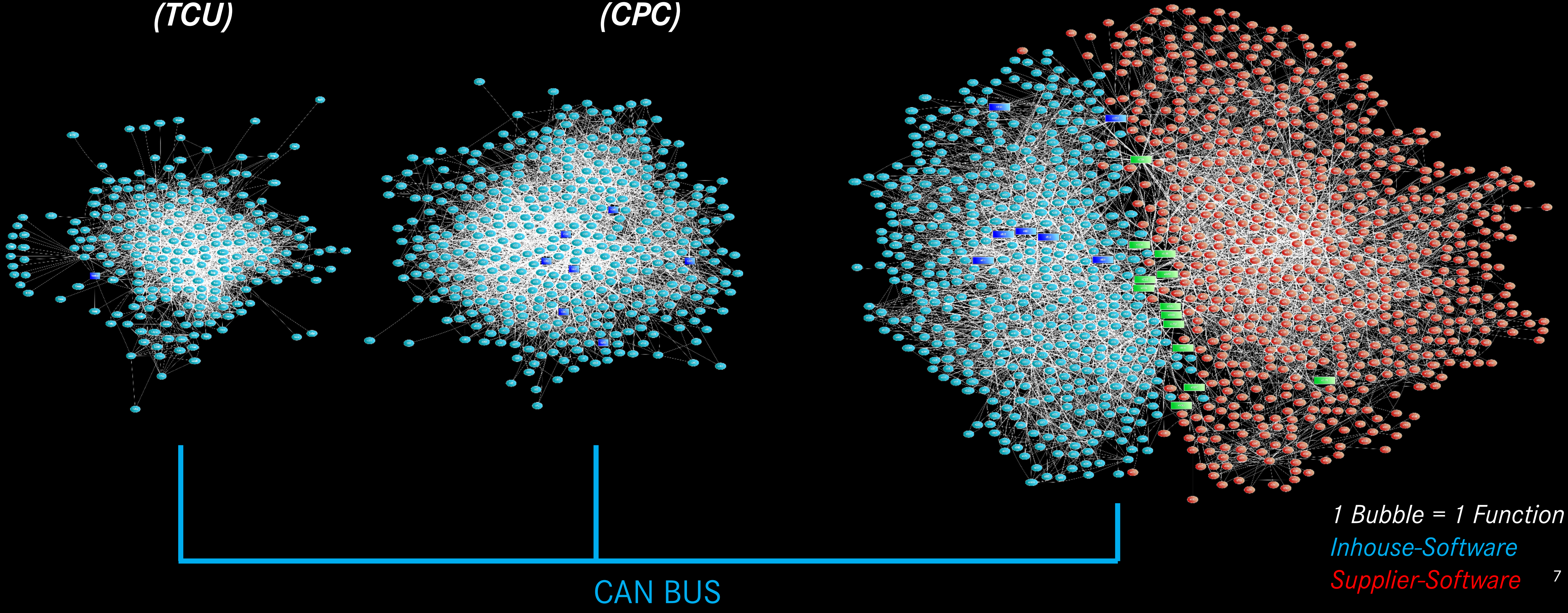


Powertrain Software

Transmission Control Unit (TCU)

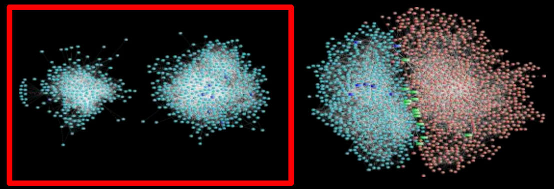
Central Powertrain Controller (CPC)

Engine Control Unit (ECU)



Completeness of powertrain software virtualization is crucial.

Creating Virtualized Control Units: Build Inhouse-SW for SiL (CPC, TCU)

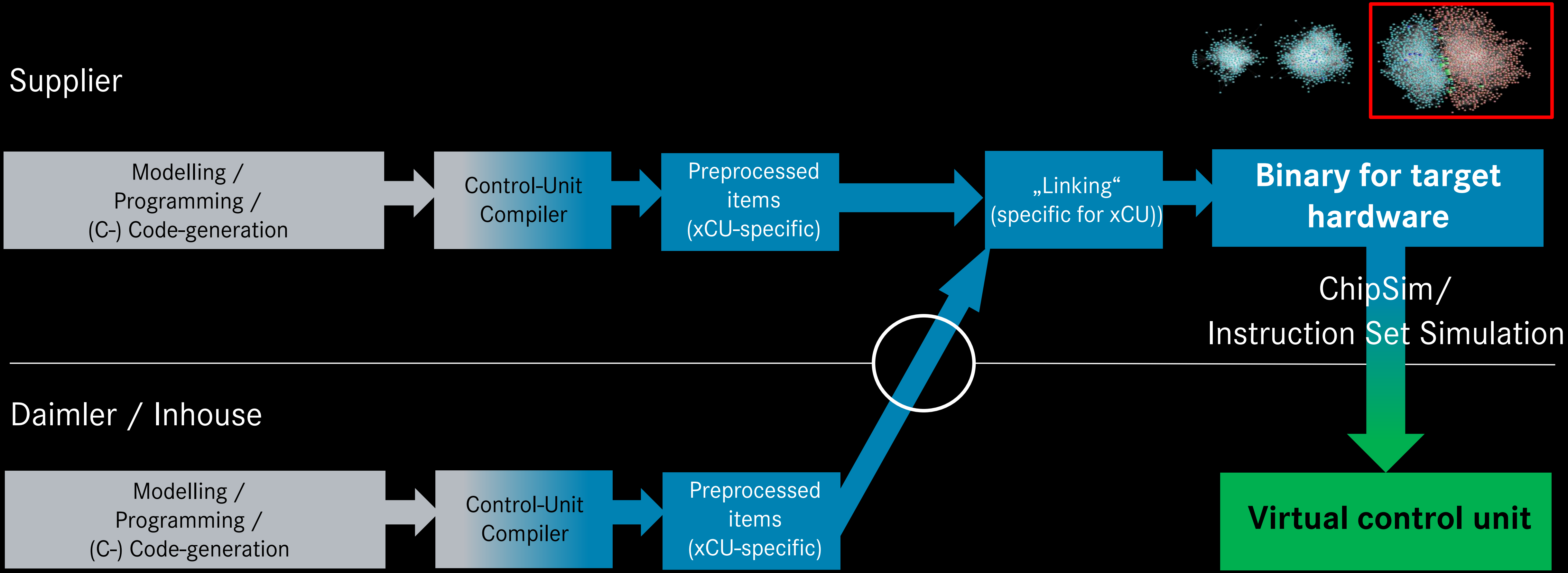


Daimler / Inhouse

Modelling / Programming / (C-) Code-generation → SiL-Compiler → Preprocessed items (object files, SiL-specific) → „Linking“ (SiL-specific) → **Virtual control unit**

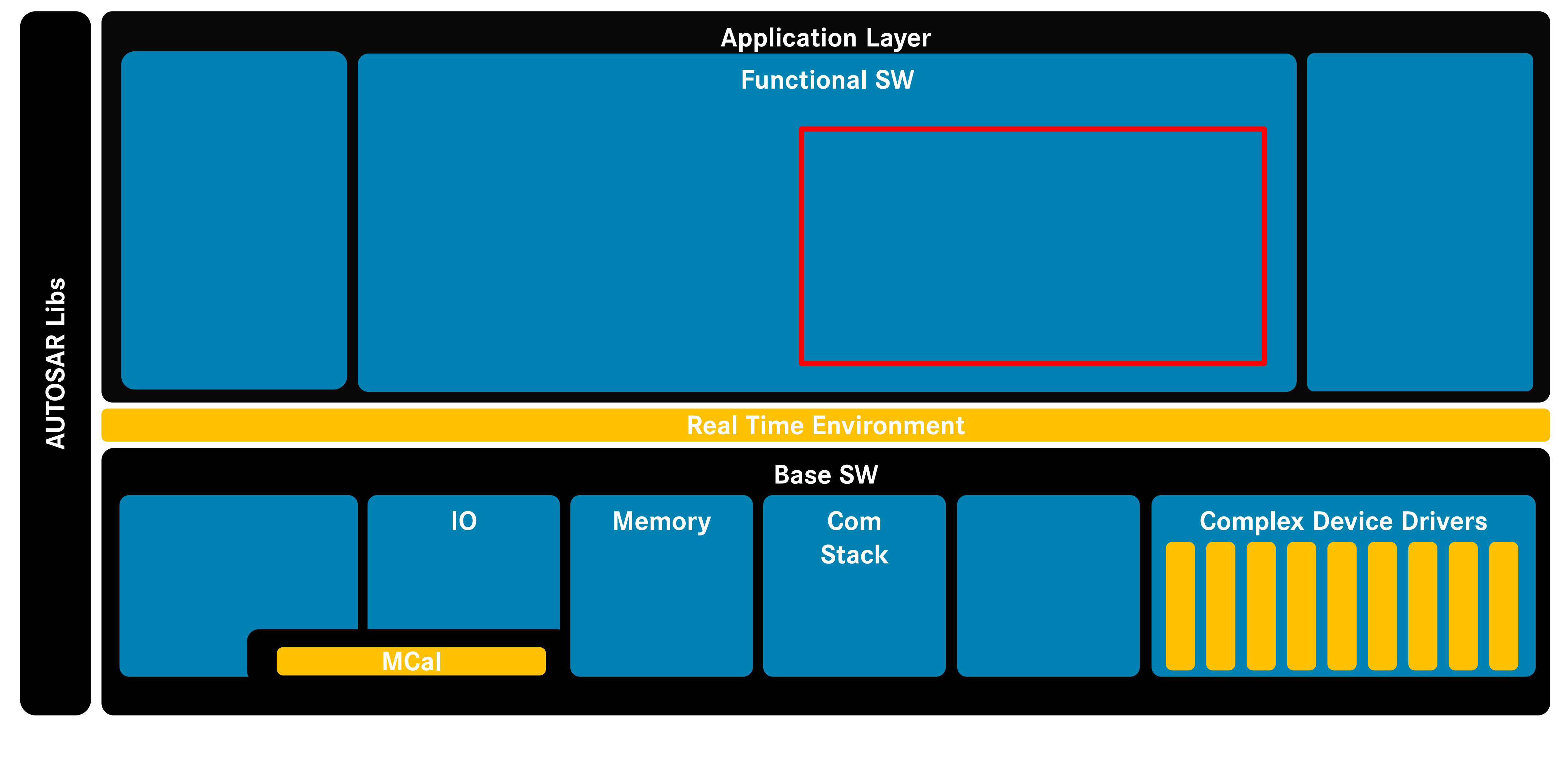
Inhouse Software: Direct usage on standard PC by tailored build process

Creating Virtualized Control Units: ChipSim for Multi-source xCU

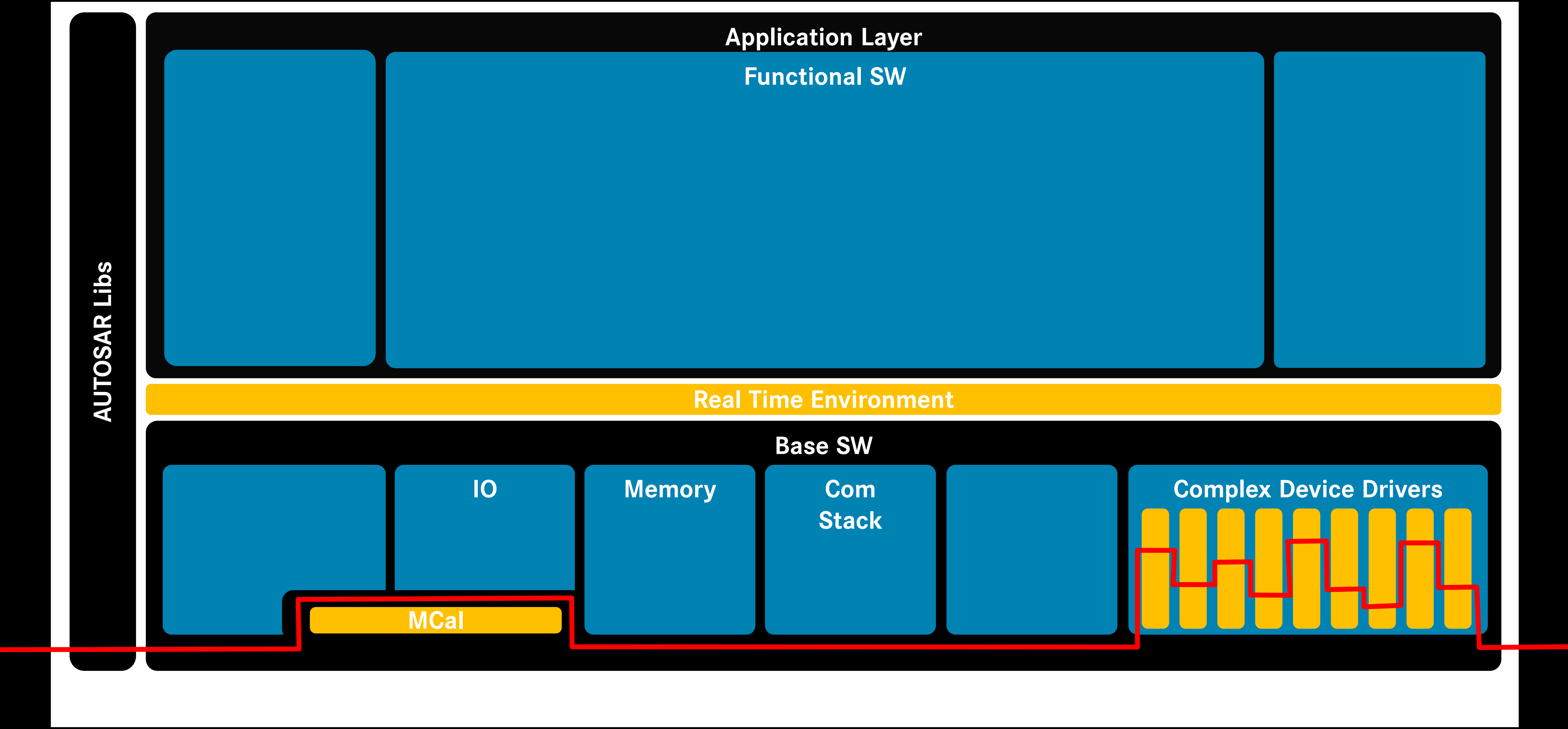


ChipSim: technology with attractive „one-size-fits-all“ attributes. Detailed at instruction set level.

Same Difference – It Does Matter How Much of the xCU Is Included!



Same Difference – It Does Matter How Much of the xCU Is Included!

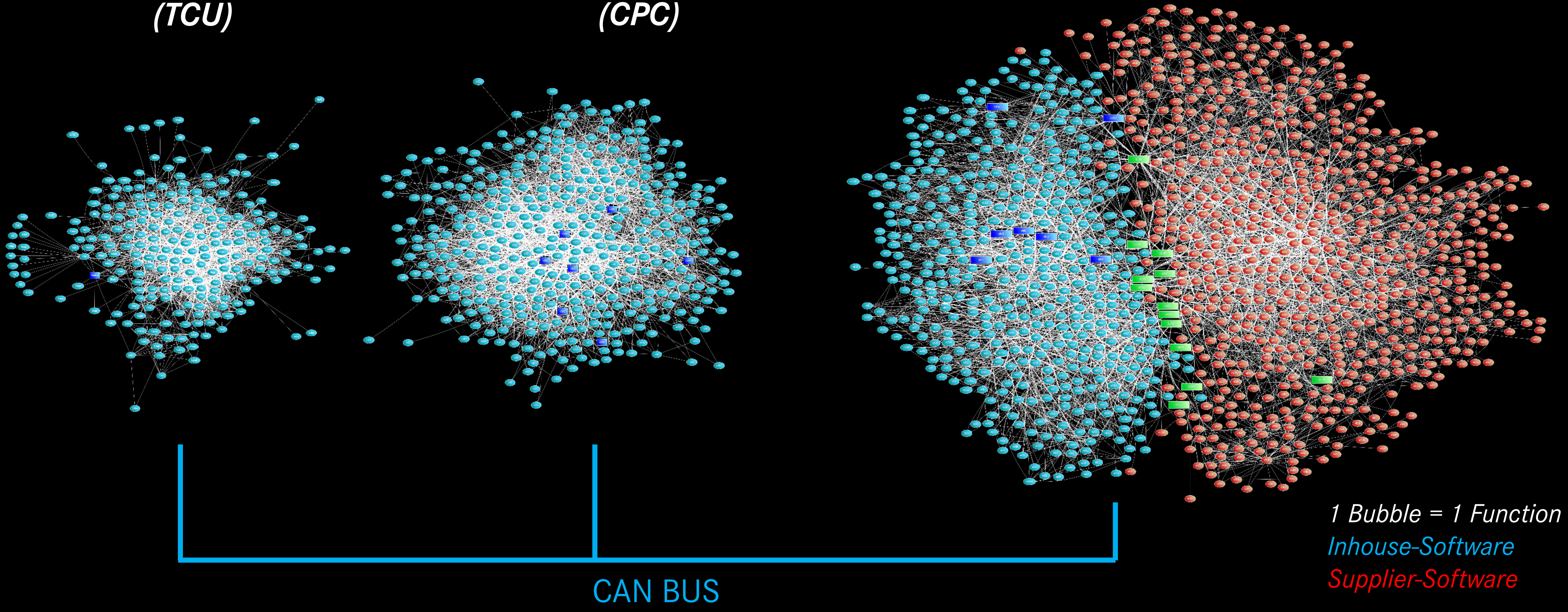


Powertrain Software

Transmission Control Unit (TCU)

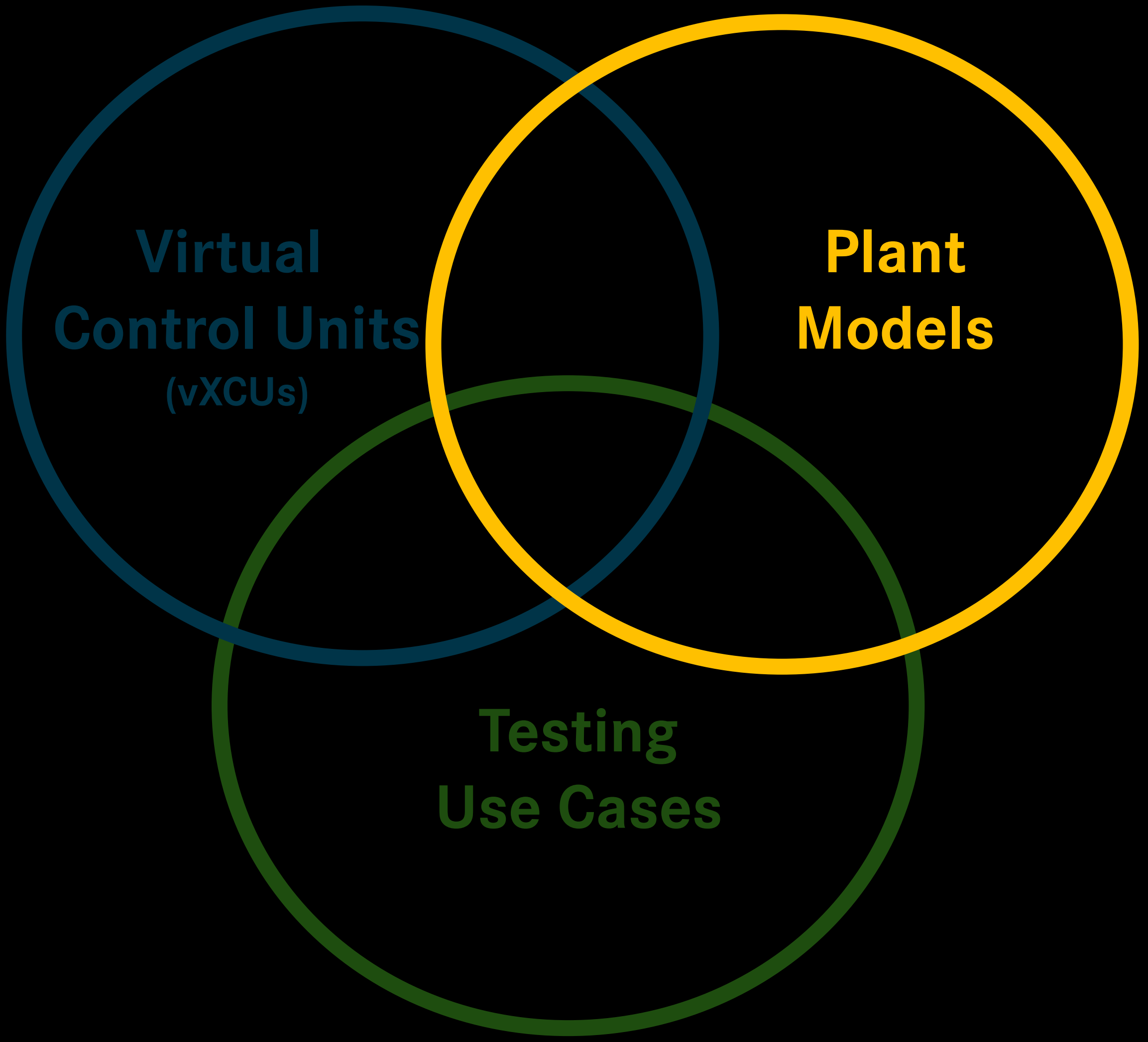
Central Powertrain Controller (CPC)

Engine Control Unit (ECU)



Completeness of powertrain software virtualization is crucial.

Ingredients

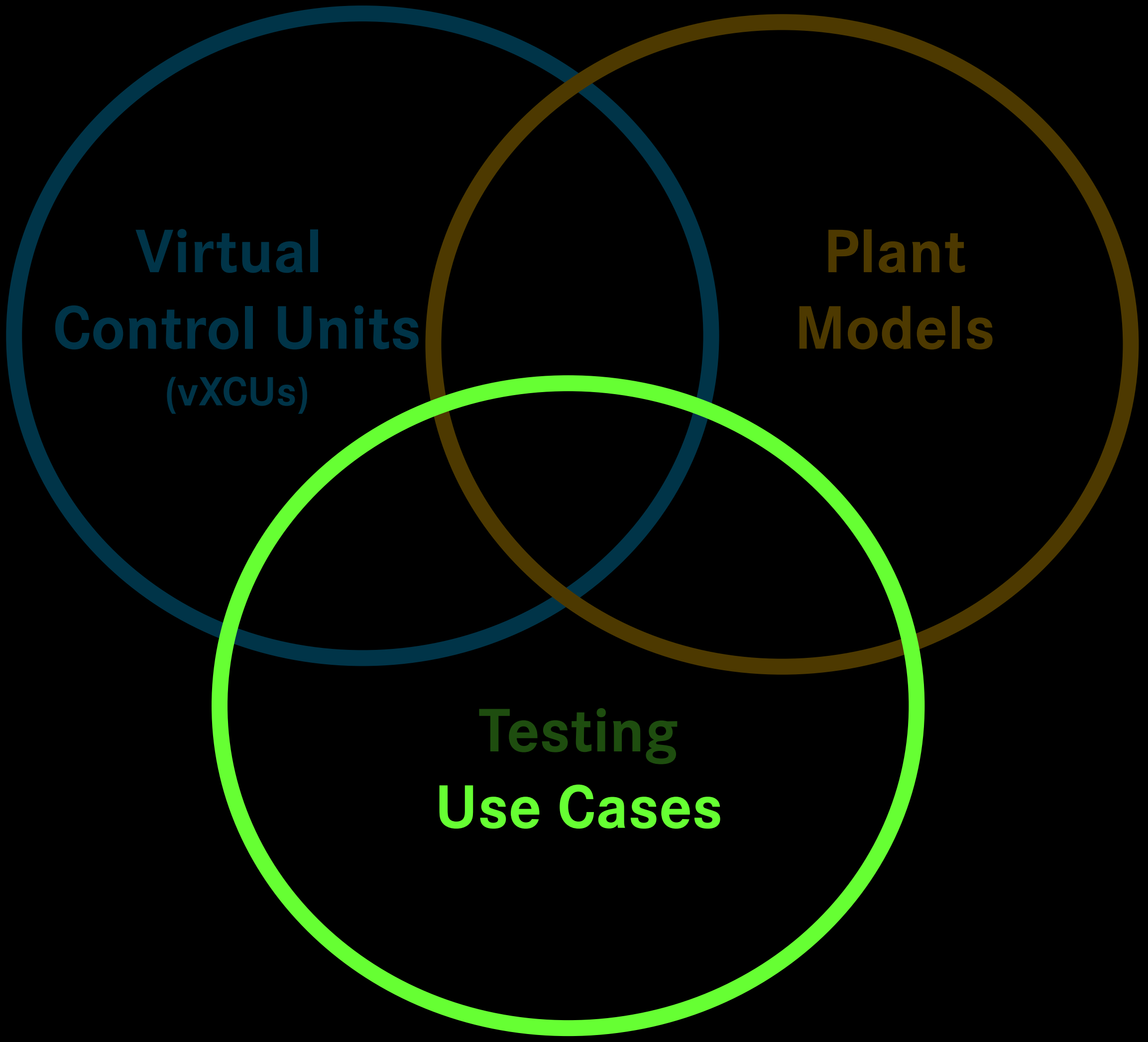


Plant Models: Quality Levels – Example: Engine Model

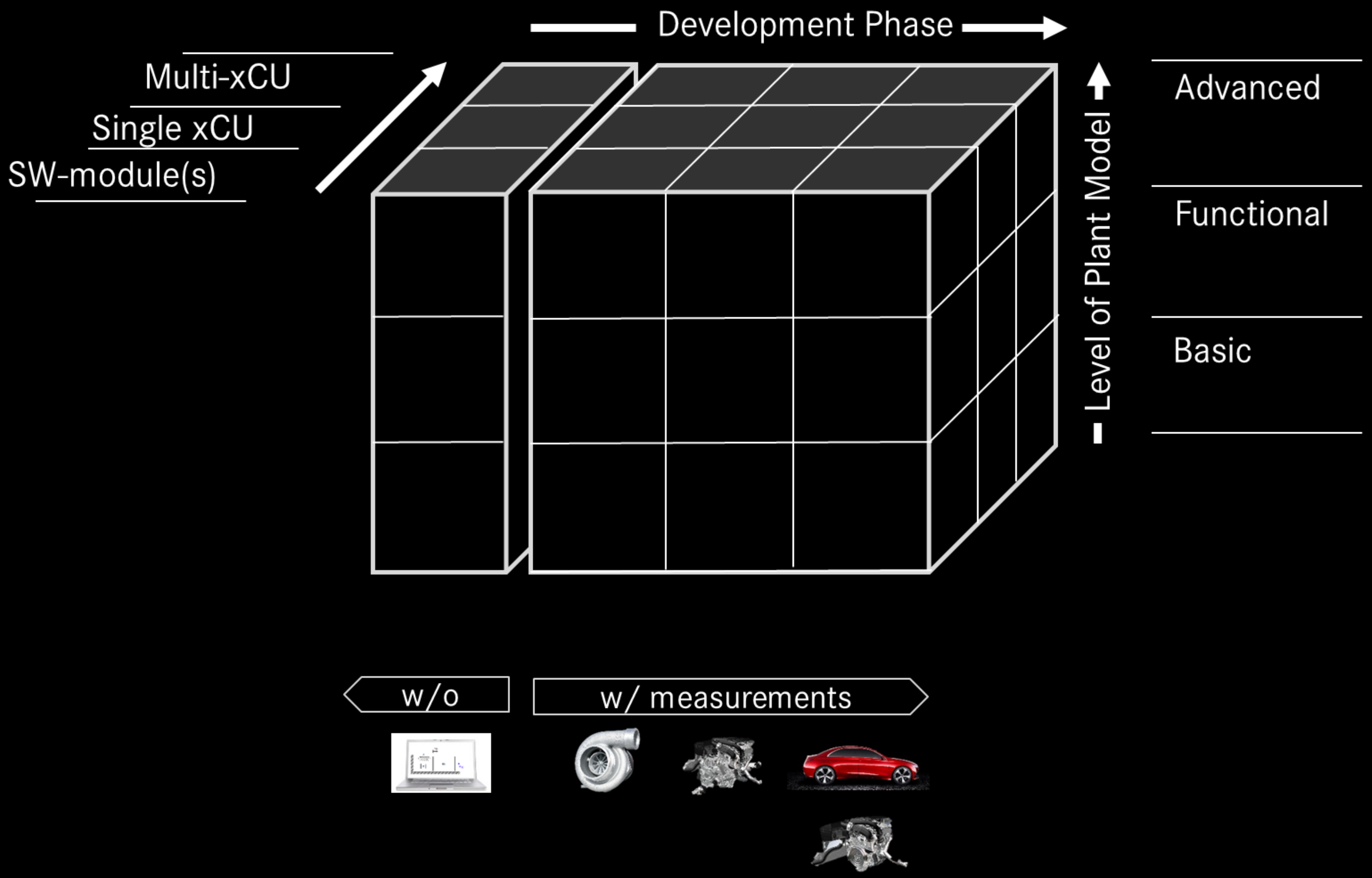
Quality level	Output e.g. engine	Characteristics	Examples	
Advanced	torque, rpm CO_2 thermal properties	• Quality and accuracy comparable to production and vehicle-to-vehicle variation • Excellent transient response matching	DEM (discrete event models, crank-angle-based models)	cost and computational burden
Functional	torque, rpm CO_2 thermal properties	• Physical phenomenology covered • Qualitative evaluation, accuracy sufficient • Limited dynamic response (EUDC-dynamics) • Physical, empirical or semi-physical models	MVM (mean value models)	
Basic		• I/O supported, general system-behavior • Simple Look-up-models and physical approach	Look-Up-Tables	

Functional model class (e.g. mean value models for engine) suitable for many applications

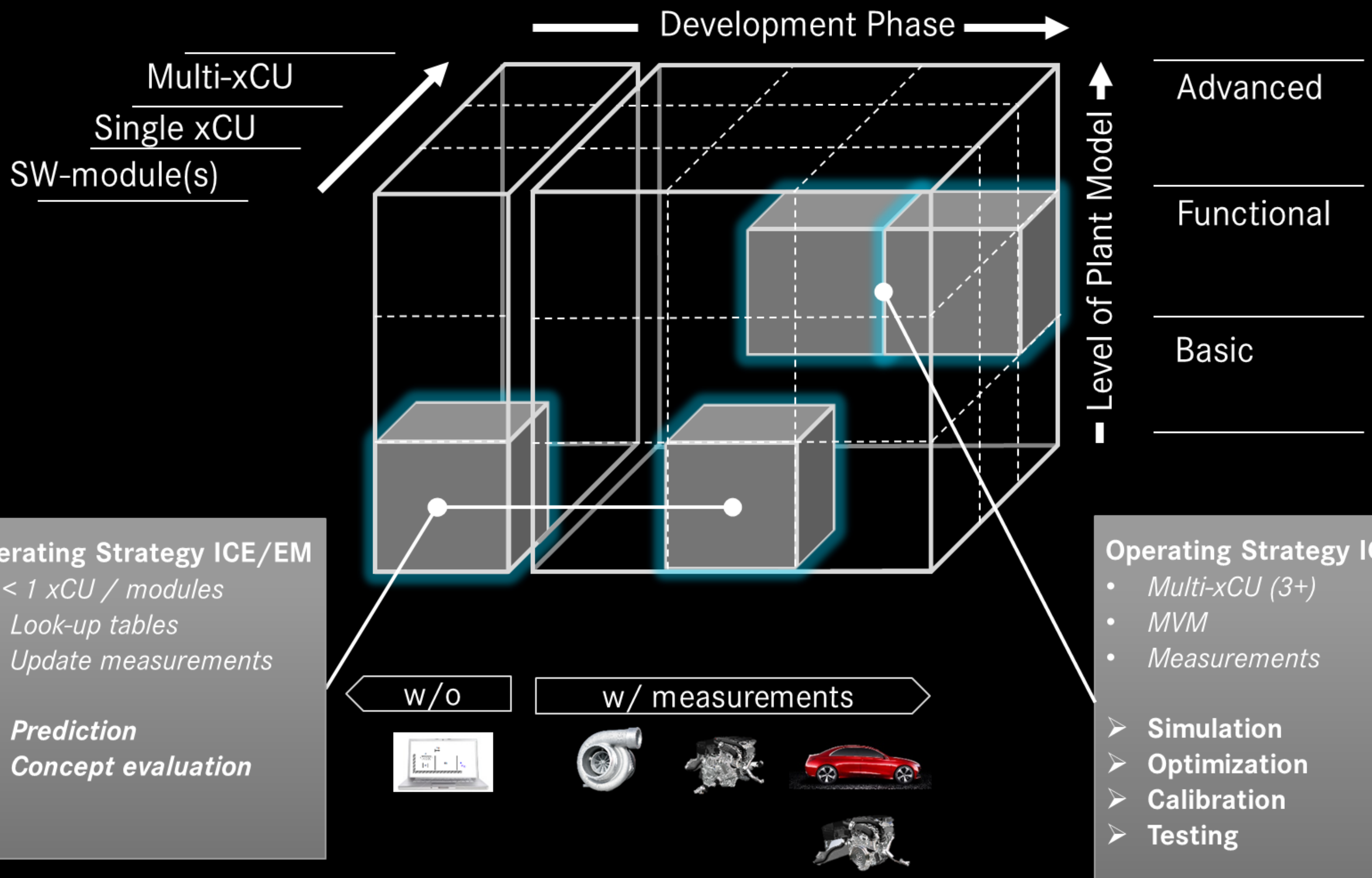
Ingredients



Use Case Matrix



Example: Operating Strategy



Operating Strategy ICE/EM

- *< 1 xCU / modules*
- *Look-up tables*
- *Update measurements*

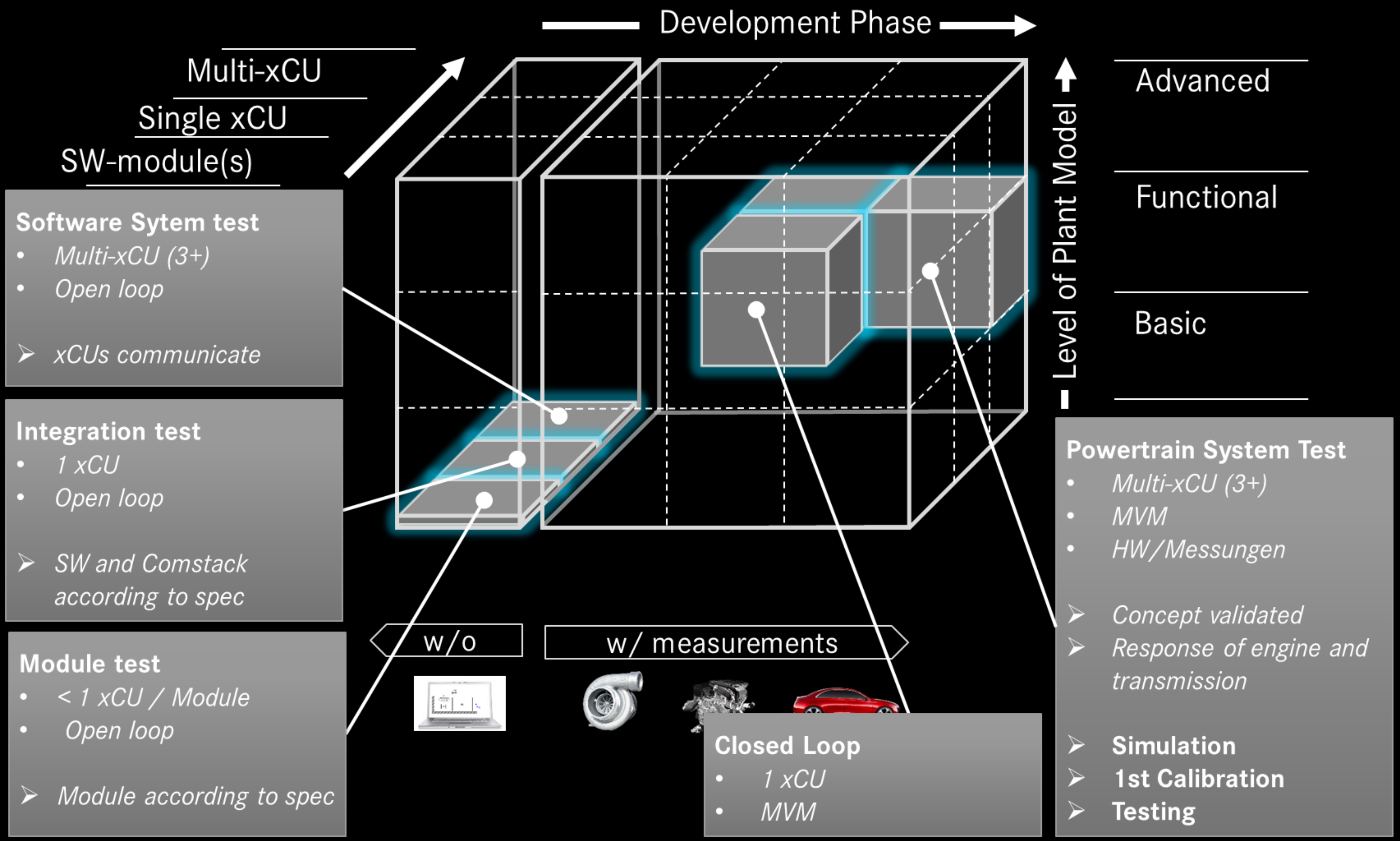
➢ ***Prediction***
➢ ***Concept evaluation***

Operating Strategy ICE/EM

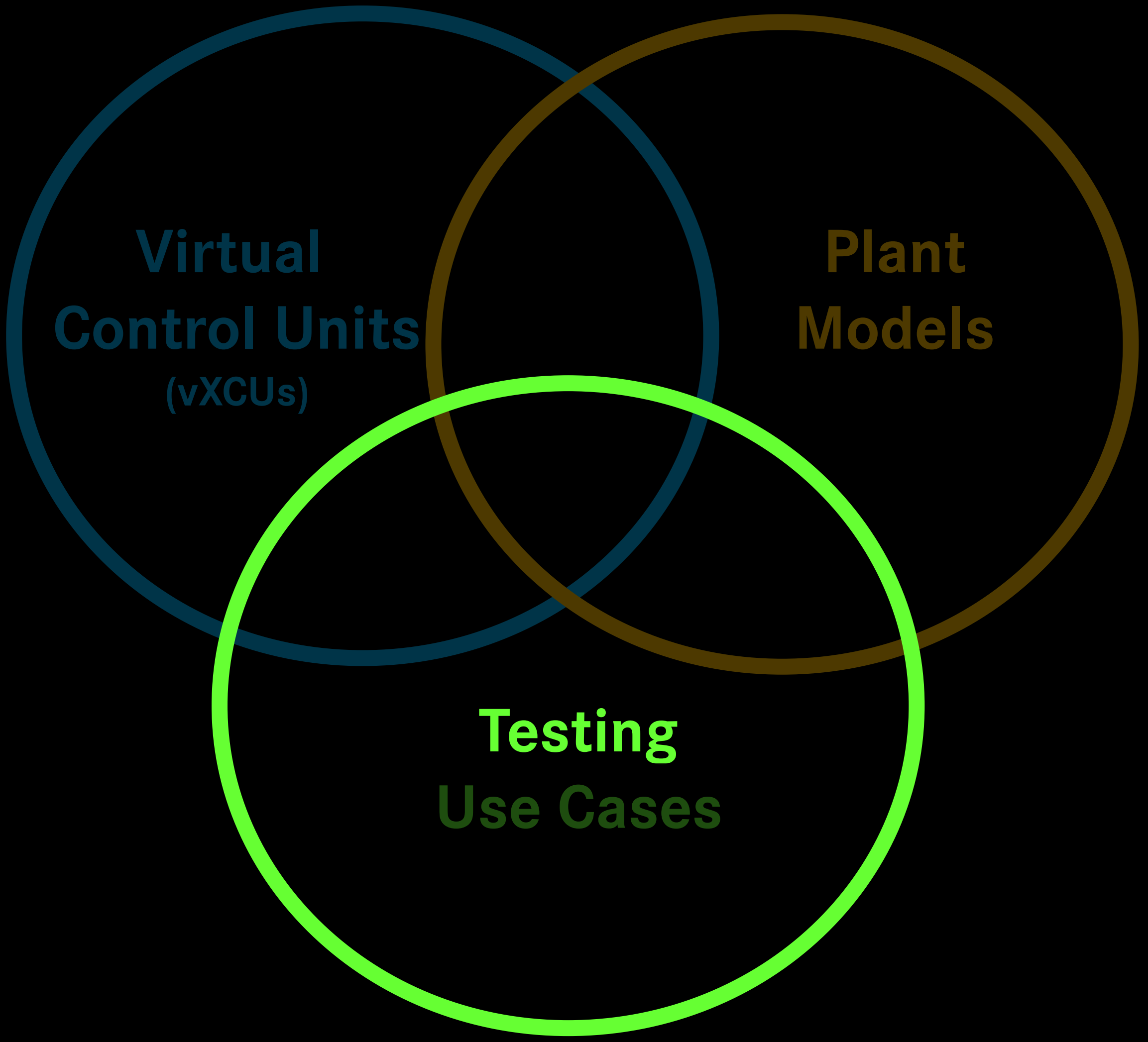
- *Multi-xCU (3+)*
- *MVM*
- *Measurements*

➢ **Simulation**
➢ **Optimization**
➢ **Calibration**
➢ **Testing**

Example: Engine Speed Governor



Ingredients



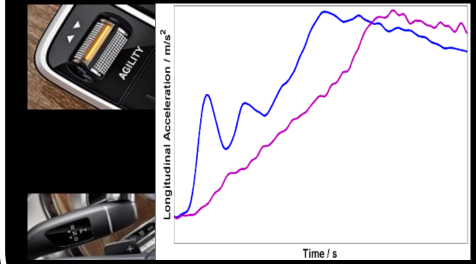
Mercedes-Benz

Testing – Test Types

Automation

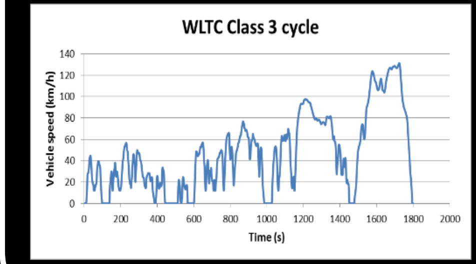
Unrestricted Driving

Drivablility Events



Objective Criteria

Test Cycles



Cumulative Evaluation

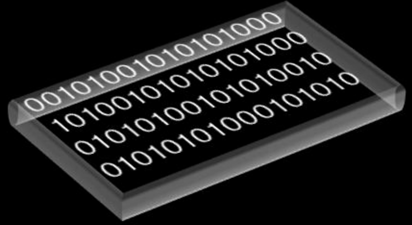
Rerun Street Tests

Robustness & Statistics

Test Case

Combinatorial Analysis

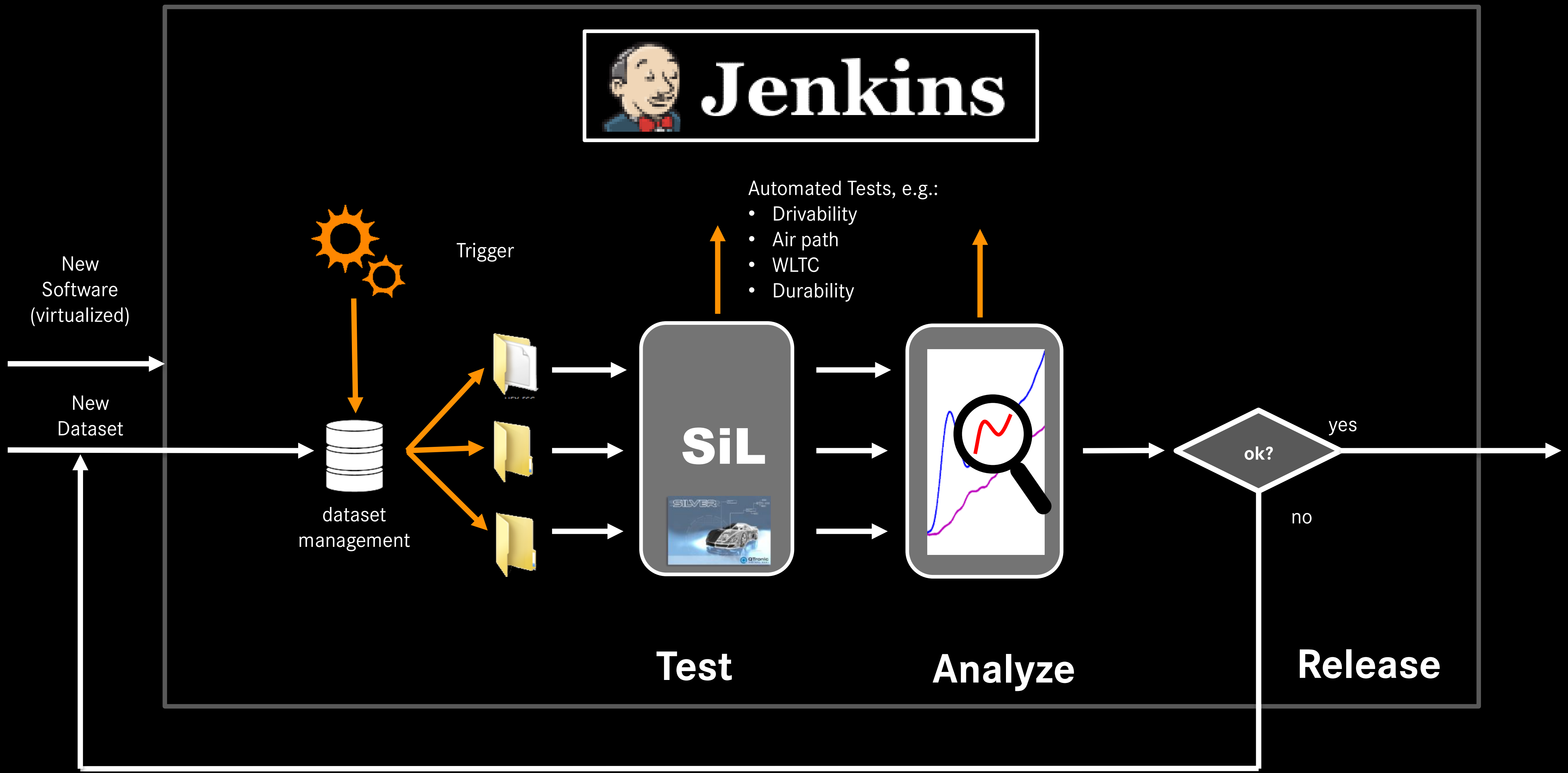
Explorative Tests



Coverage

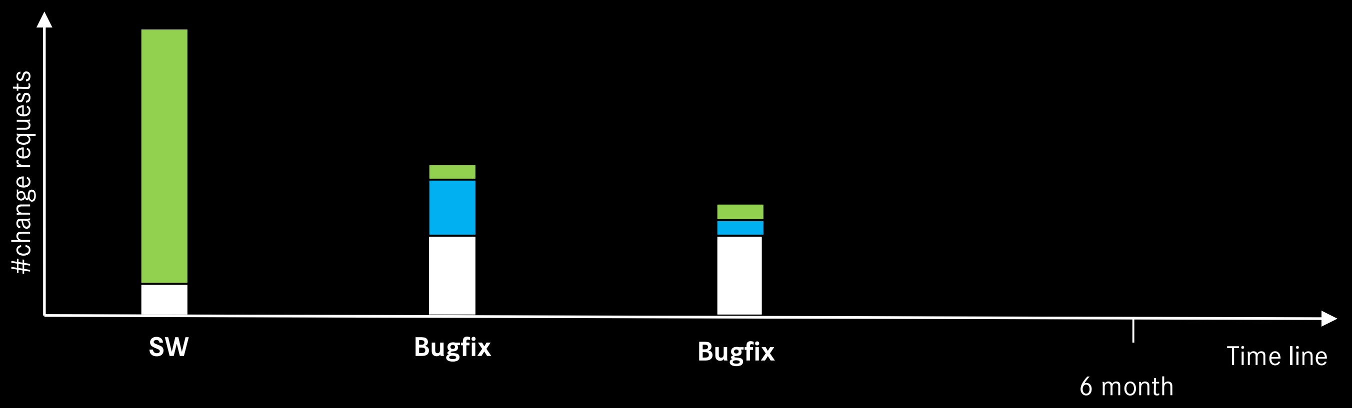
In addition to unrestricted driving, various test types can be implemented which - when used repeatedly - can be triggerd automatically.

Test Automation

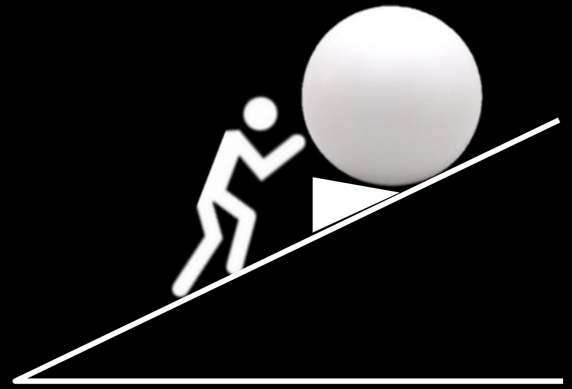


Why Automated Testing...

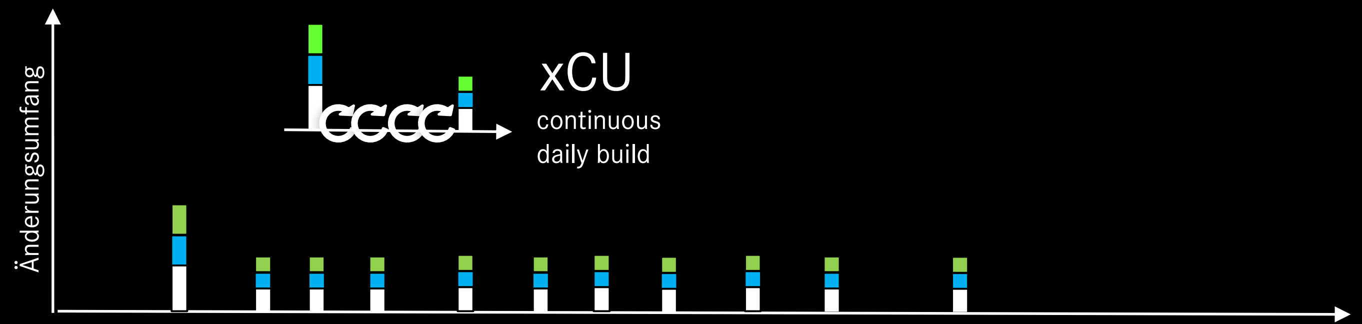
xCU
V-model
classic
retro



new
Specfixes
Bugfixes

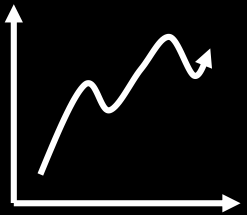
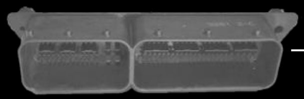


xCU
Agile
Sprint - Release
xCU



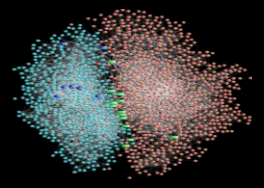
Test automation enables permantent evaluation of the development progress and supports continuous improvement of software and calibaration.

Aspects for Assessing a Software-in-the-Loop Environment

Completeness		*Completeness is more important than specific level of detail*	*No limitations, no pitfalls, interactive operation*		**Unrestricted Driving**
Open Models		*Open models vs. blackbox*	*One task → one tool for measure-ment, calibration and diagnosis*		**Standard tools**
Focus on Physics		*Physical phenomena more important than accuracy*	*Rights Management (DRM) – Content defines value*		**Access Rights**
Flexible Licensing		*Royalty-free models*	*Pure Software – always available & highly scalable*		**“To-go”-Flexibility**
Modularity and Extendability		*Intuitive interfaces, standardized exchange formats*	*A simple user interface increases acceptance*		**User-friendly**

Conclusions on Business Model

Control Modules



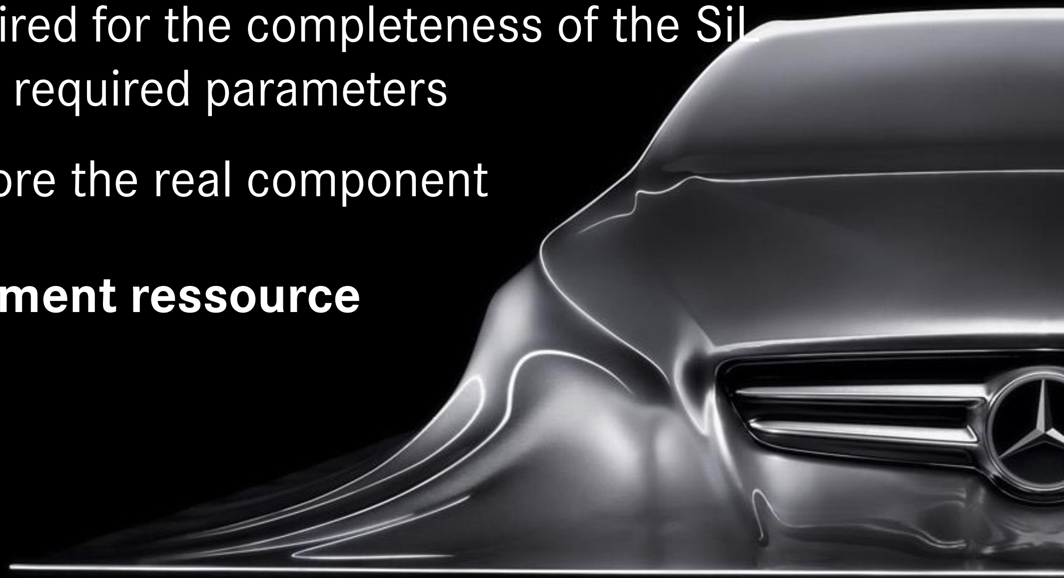
- The vXCU is part of the delivery and part of the business model OEM – supplier
- Technology for XCUs with software from multiple sources available
- The virtual XCU must be available before the real one or the software

⏩ **If it has software, it has a virtual control module**

Plant Models

- Supply of plant models which are required for the completeness of the Sil (e.g. battery, starter/alternator) or the required parameters
- The plant model must be available before the real component

⏩ **The plant model is the first development ressource**



Conclusions on Business Model

Exchange Format

- Plant models and vXCUs should not be restricted to a tool-specific format
- Exchange between multiple model owners required
- Enabler for completeness of SiL

⏩ **Independence from target platforms by means of exchange formats**

Standards

- Strong alignment to standards, both "external" (e. g. FMU) and "internal" (e. g. for open and standardized software architecture AutoSAR)

⏩ ***„Comply on standards, compete on implementation"***

Conclusions on Business Model

Skills

- Development on virtual vehicles becomes an integral part of every development discipline
- Simulation provides the modeling skills but not (all) the simulation results
- Skill set part of the training-on-the-job for professionals or the academic education

⏩ **Additional skills for virtual development**

Engineering Tools

- Tools to analyze and post process test results to be activated by generic test automation and reporting

⏩ **Compatible with both, real and virtual test environment and process**

Summary

- The goal is to use a virtual vehicle for development where it is suitable.
- For powertrain development, e.g. for drivability calibration, this requires that all control units and powertrain subsystems are virtualized. Completeness of the system is crucial - hence, powertrain system simulation.
- The key technology is the Software-in-the-Loop approach for digitalizing control systems.
- The virtualized control modules are integrated with powertrain plant models using a co-simulation tool.
- The SiL-platform provides engineers with the complete application software for the ECUs and all relevant powertrain models. It is used for software development, calibration and simulation tasks.
- The virtual platform catalyzes automated testing.
- The option of integrating powertrain system simulation into development project has an impact on the cooperation of OEMs, suppliers and engineering companies.

Thank You for Your Attention!