Models for
Automotive Embedded Systems Development

Henrik Lönn, Technology Specialist, Embedded Software Development, AB Volvo.

Volvo Group Business Areas

VOLVO TRUCKS

RENAULT TRUCKS

MACK TRUCKS

UD TRUCKS & JVs

ARQUUS

CONSTRUCTION EQUIPMENT

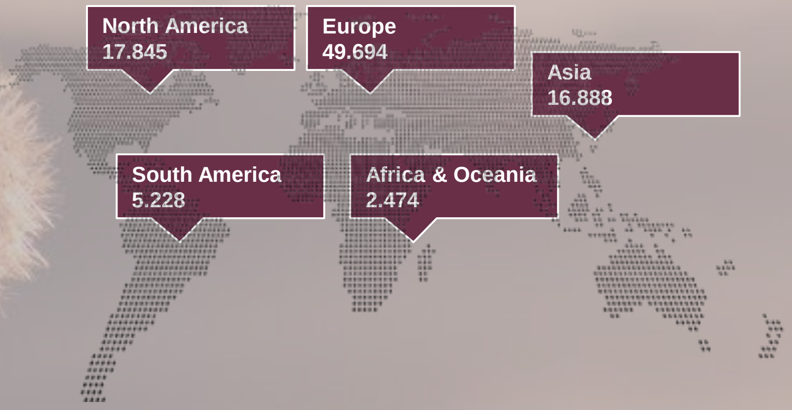
BUSES

VOLVO PENTA

VOLVO AUTONOMOUS SOLUTIONS

VOLVO FINANCIAL SERVICES

Our organization



Volvo Group CEO
& Group Executive Board

GROUP FUNCTIONS

- Group Functions

BUSINESS AREAS

- Renault Trucks
- Mack Trucks
- UD Trucks
- Volvo Trucks
- Volvo CE
- Volvo Buses
- Volvo Penta
- Volvo Financial Services
- Volvo Autonomous Solutions
- Arquus

TRUCK DIVISIONS

- Group Trucks Technology
- Group Trucks Operations
- Group Trucks Purchasing

Innovative technology offerings

FOR OUR CUSTOMERS AND THE SOCIETY

AUTOMATED MANUAL TRANSMISSION

RENAULT INFOMAX

REMOTE PROGRAMMING OVER-THE-AIR (OTA)

VOLVO DYNAMIC STEERING (VDS)

WAVE PISTON FOR BETTER EFFICIENCY

PEDESTRIAN AND CYCLIST DETECTION SYSTEM

AUTOMATIC TRACTION CONTROL

PREDICTIVE CRUISE CONTROL (I-SEE)

COLLISION WARNING WITH EMERGENCY BRAKE

MACK ® GUARDDOG ® CONNECT

Concepts and technologies for the future

ACCELERATING THE TRANSITION TO A SUSTAINABLE FUTURE

AUTONOMOUS REFUSE TRUCK FOR URBAN ENVIRONMENT

AUTONOMOUS TRUCK FOR MINING

VOLVO CONCEPT TRUCK

VOLVO VERA

ELECTRIC ROAD SYSTEMS

FULLY ELECTRIC CONSTRUCTION EQUIPMENT WITH AUTOMATION

3D PRINTING FOR LIGHTER ENGINES

WIRELESS SENSORS TO REPLACE CABLES

PLATOONING

AND MANY MORE

Take Home Messages:

- Appropriate Software Architecture is Instrumental
- Software Architecture Competence is Critical for Swedish Industry
- Modeling is an Enabling Technology for Continuous Integration of Complex Products

Product Challenges

- Reduced Lead Times
- Continuous Deployment
- Continuous Integration
- Quality
- Reuse

Process Challenges

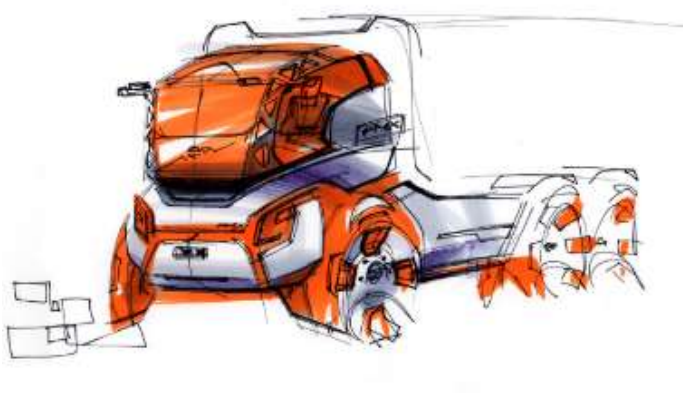
- Multiple cycle times
- Multiple Aspects
- Multiple Domains
- Multiple Teams

- Automation
- Connectivity
- Electrification
- Control Strategies
- Architecture (re)volution

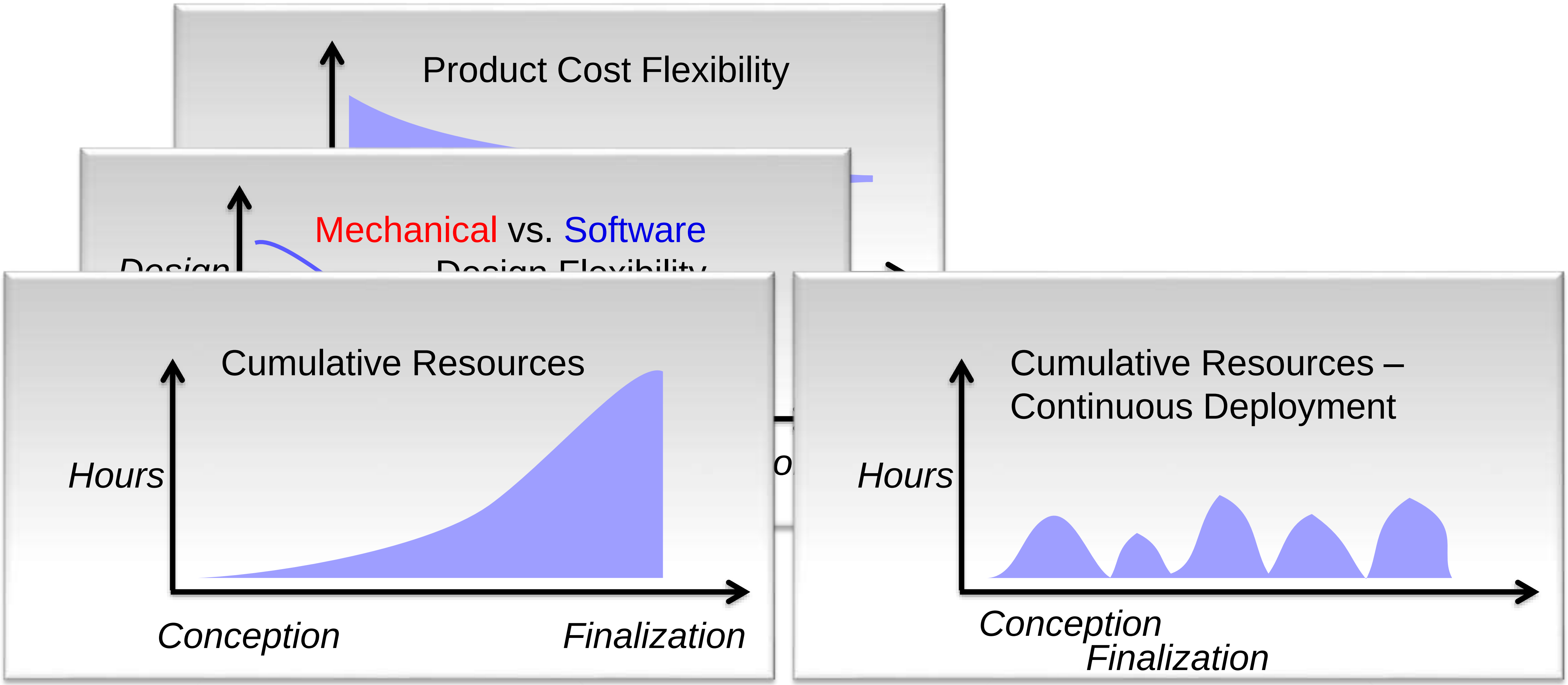
Why

What

How



Get Things Right - Continuously



Example: Testing

Example: Road and Track Testing

- When Simulation is too difficult
- When Simulation is Impossible
- Because it is “Free”

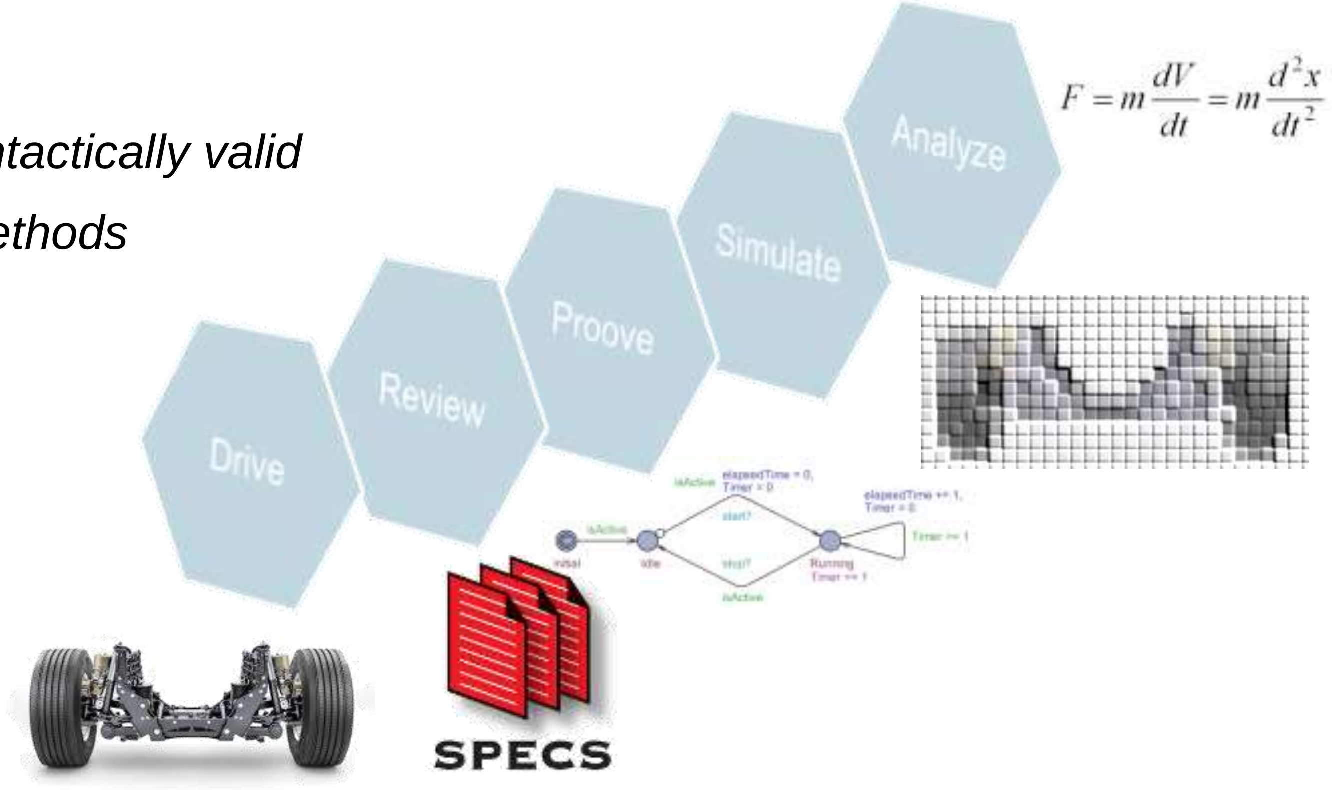
Example: Analysis and Simulation Based "Testing"

- To exercise rare situations
- To exercise dangerous situations
- To accelerate time
- To accelerate coverage
- To be repeatable
- To allow debugging
- To cover large sets of configurations
- To cover fictuous configurations

In General: Multi-Method

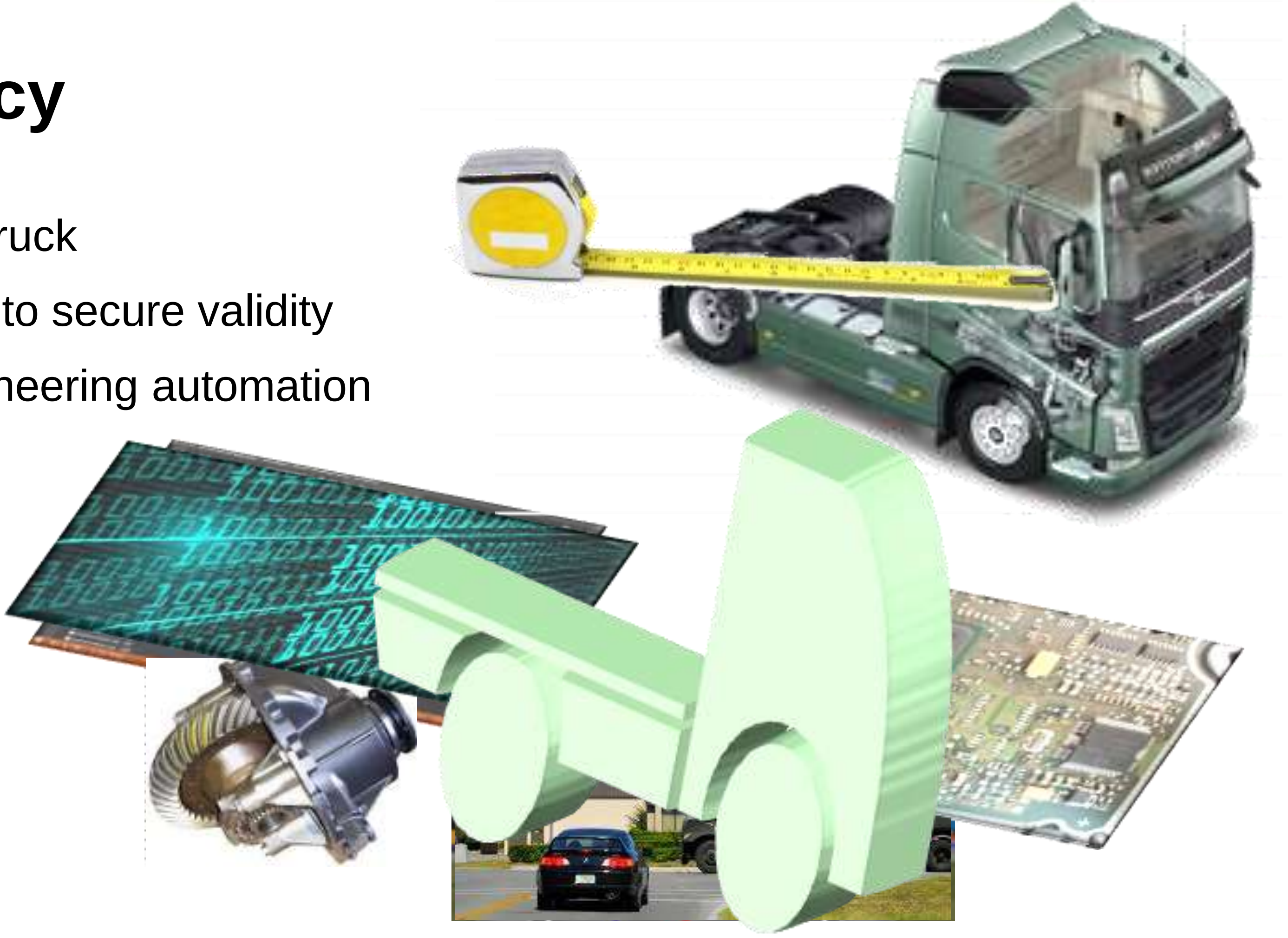
Multiple Engineering Activities

- *Input must be semantically and syntactically valid*
- *Input must be consistent across methods*



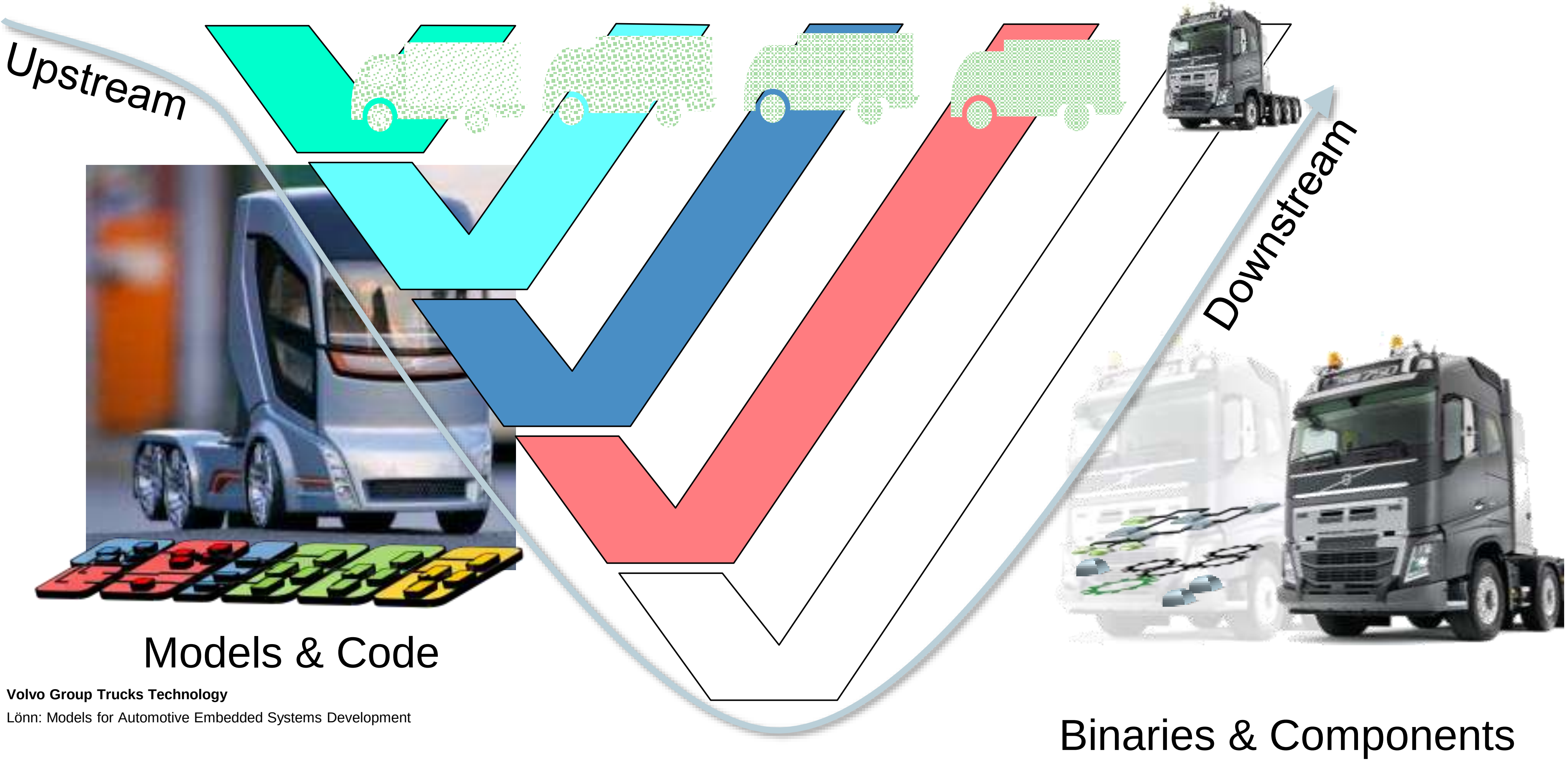
Consistency and Accuracy

- Models are a representation of the real truck
- Data from factory, track, field is required to secure validity
- Models allow engineering rigor and engineering automation

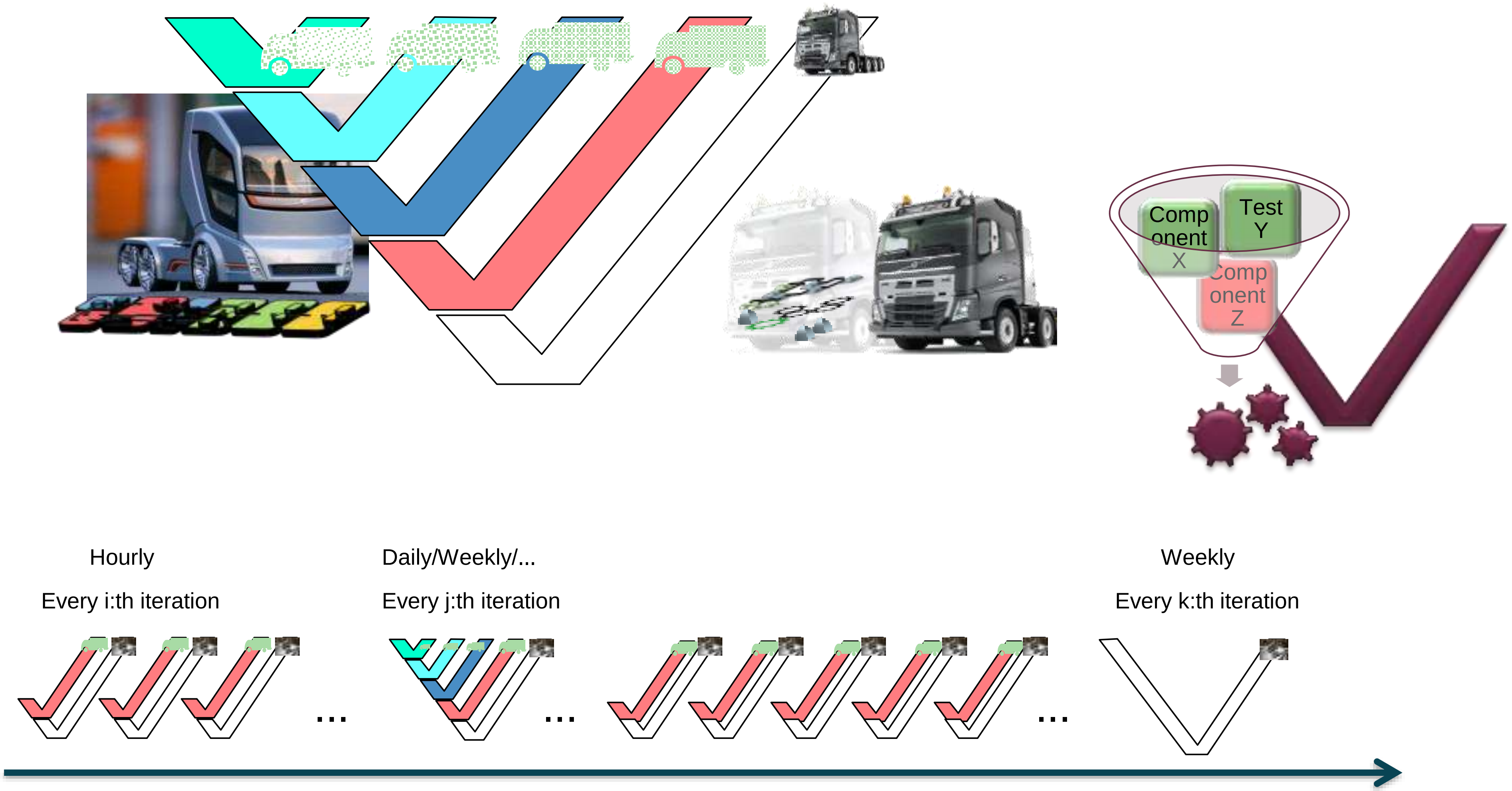


Engineering Information and Activities

V model as reference
…not to prescribe timing and sequence…



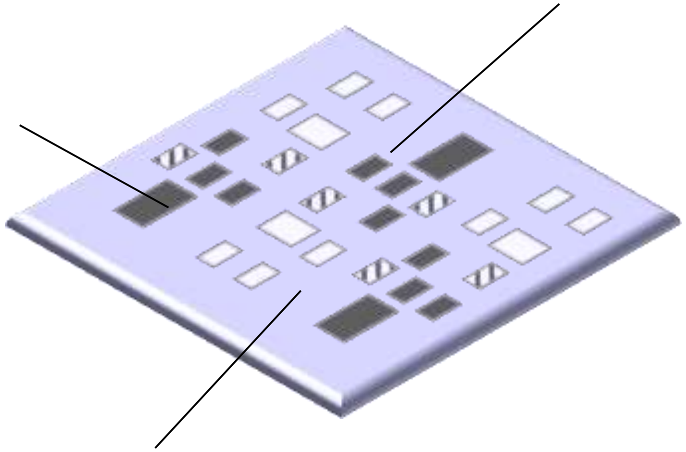
V model vs. Continuous Integration and Delivery



Collaborate Efficiently

- Product complexity call for advanced models and tools
- Modelling is a significant Effort

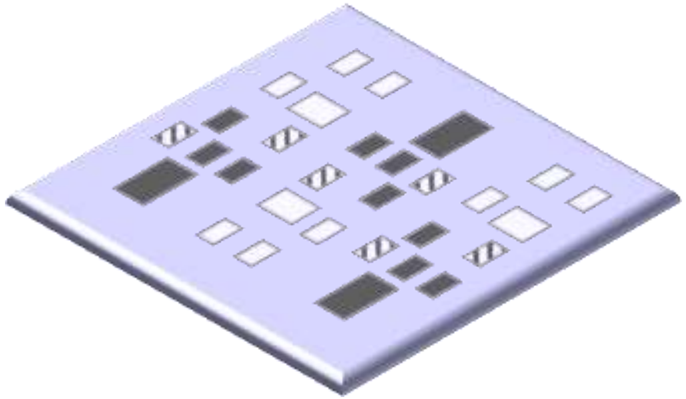
Vehicle Manufacturer must integrate models



Control over Engineering Information

- Engineering documentation is a core asset
- Increased use of models represents risk
  - Vendor Lock-in
  - Complex Information
  - Complex access to information

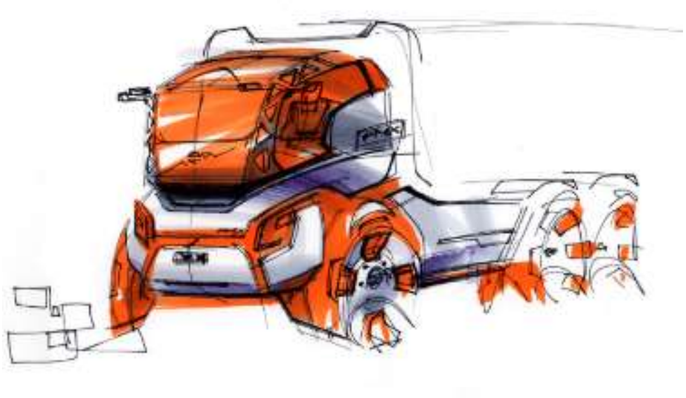
Vehicle Manufacturer must have control of their models



Why

What

How



Vehicle Content

Solution independent description of vehicle

”Features” to decompose content

Height Control

Climate Control

Speed Keeping

Steering

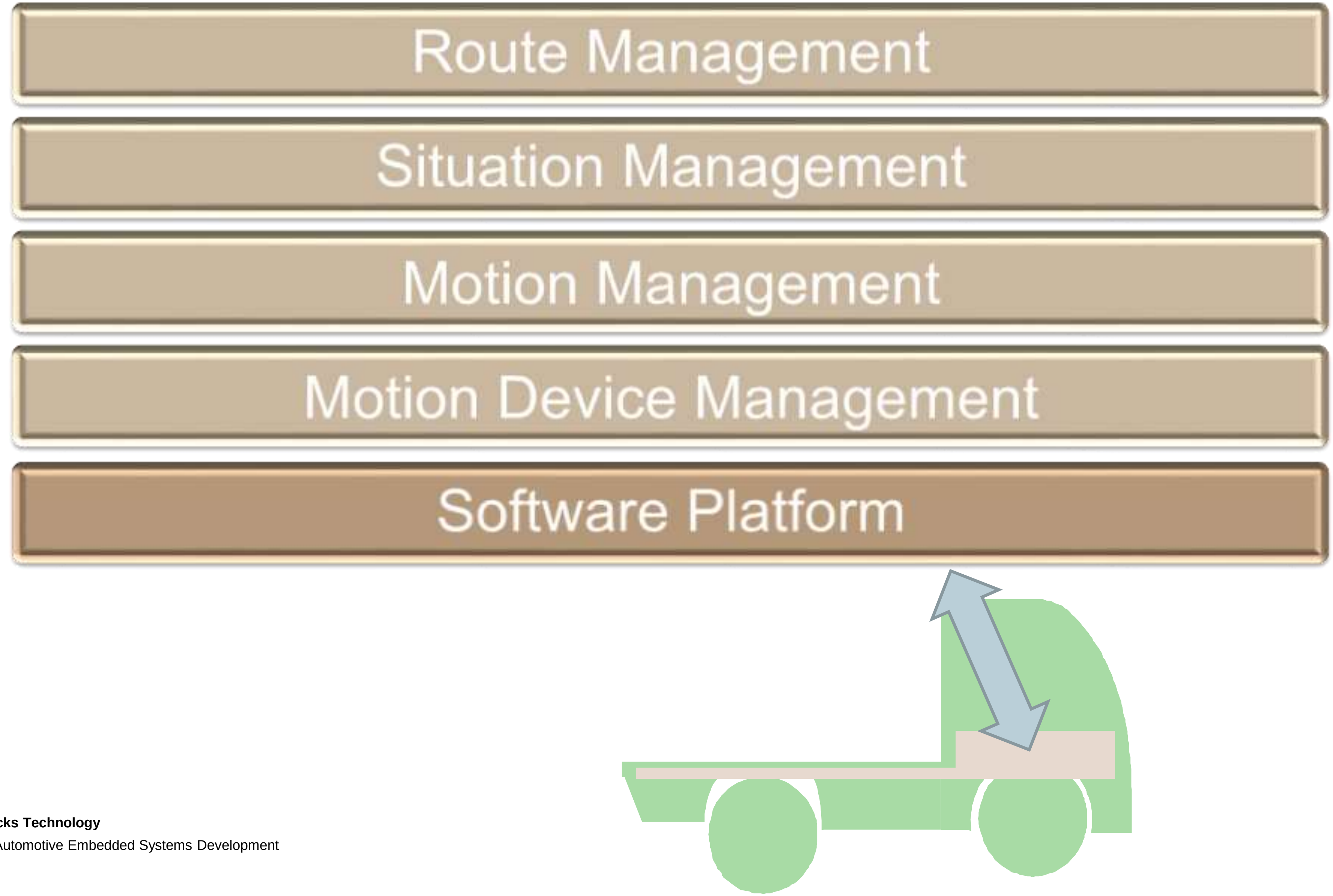
External Lights

Entertainment

Entertainment

Application Software Architecture - Layering

Application Software Architecture - Layering



Virtual Engineering

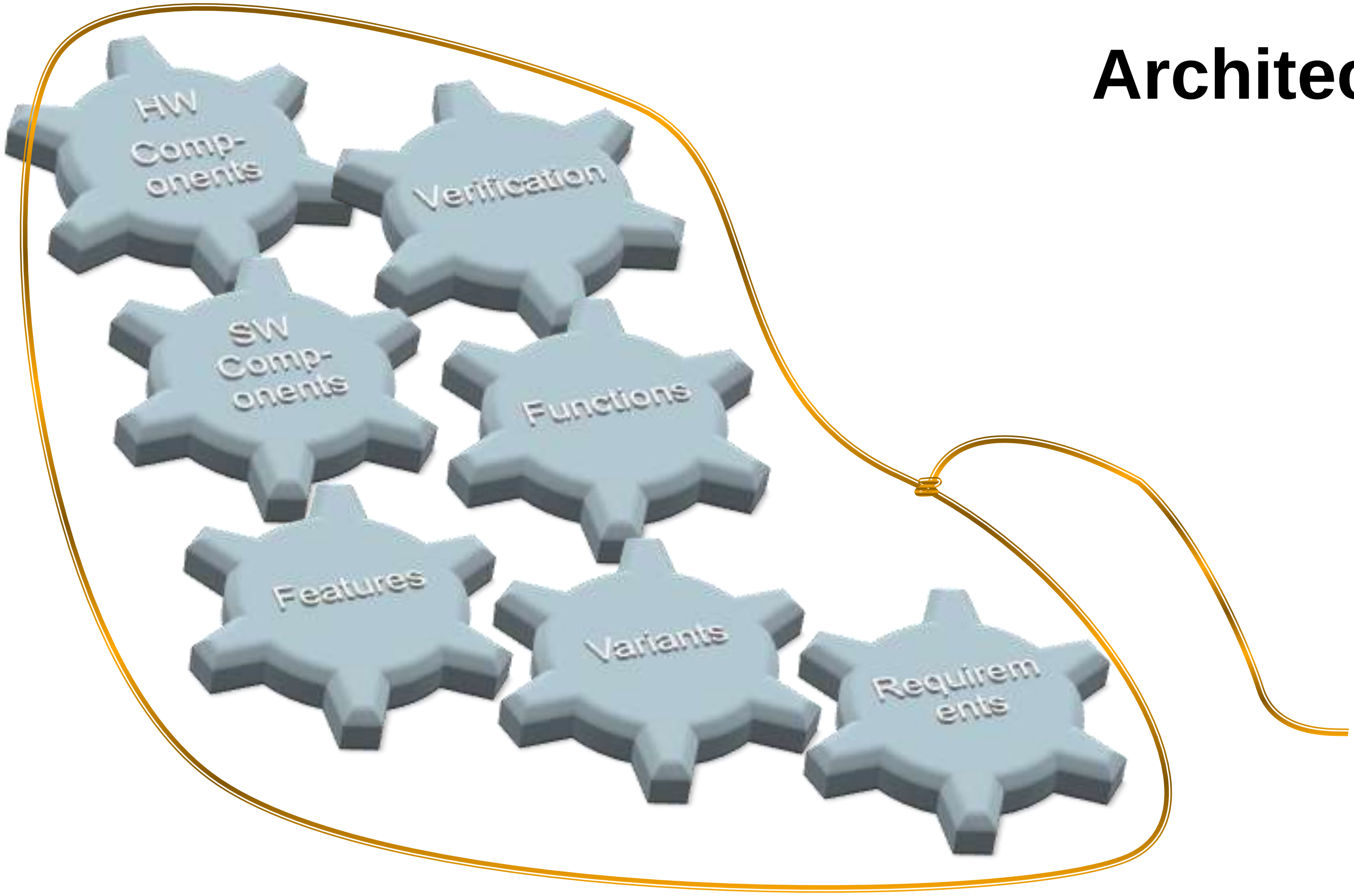
...to allow

- Instant Deliveries
- Maximize Verification Confidence
- Exercise Dangerous and Rare Events

Virtual Integration and Simulation

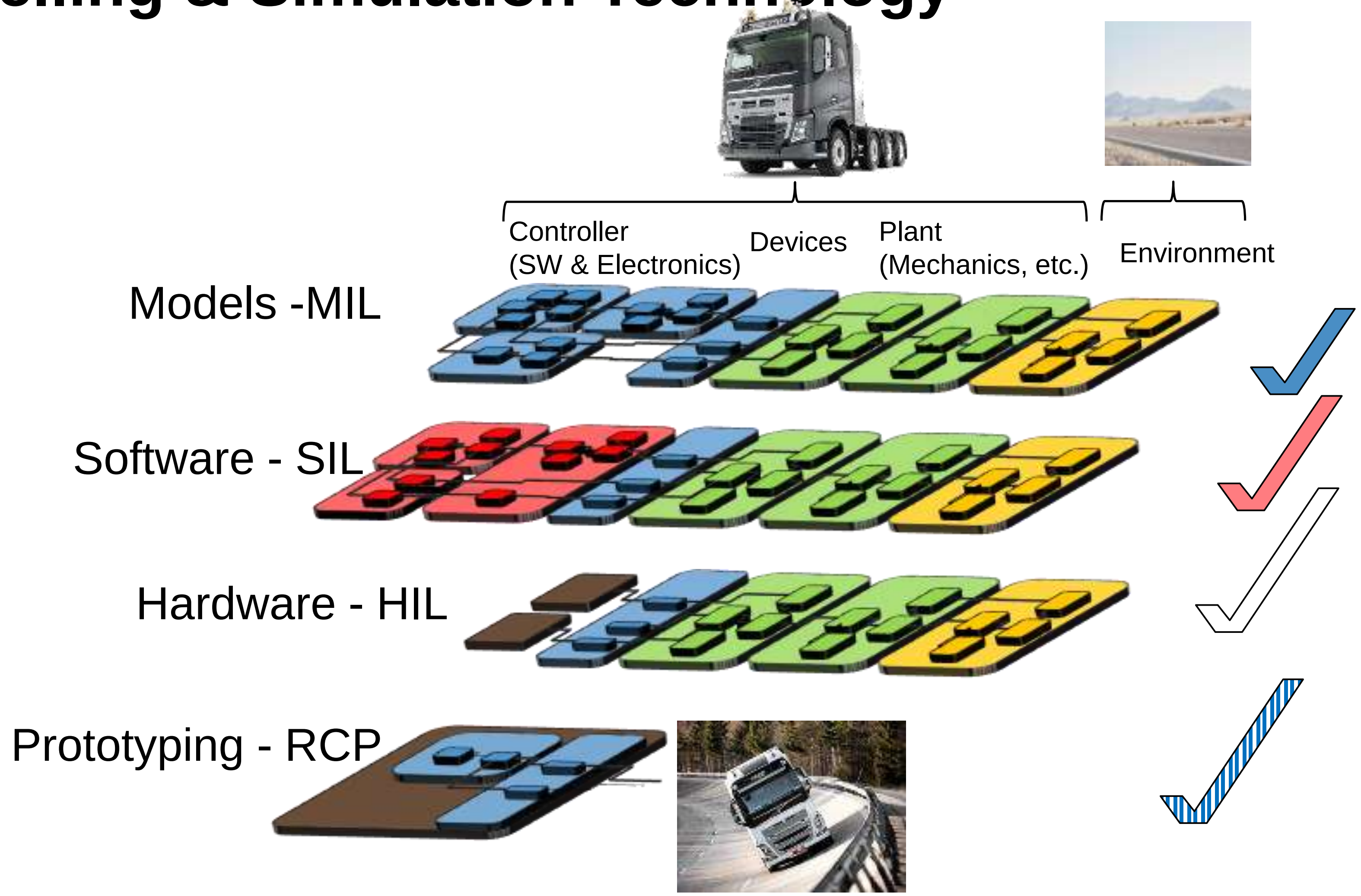
- Simulation requires models of all relevant functional aspects
  - (Software)
  - Software platform
  - Electronics
  - Mechatronics
  - Physics
  - Environment

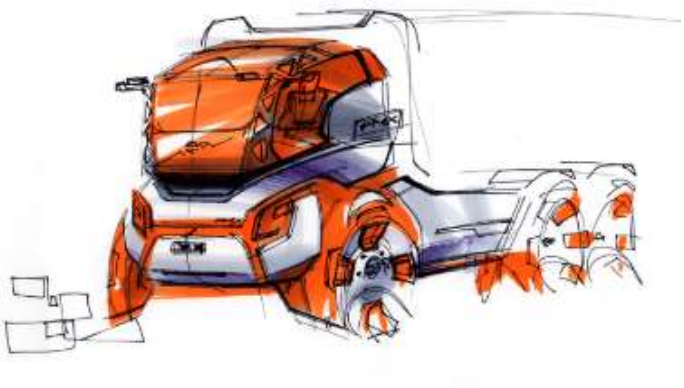
Architecture Description Language



An information model that captures engineering information in a standardized way

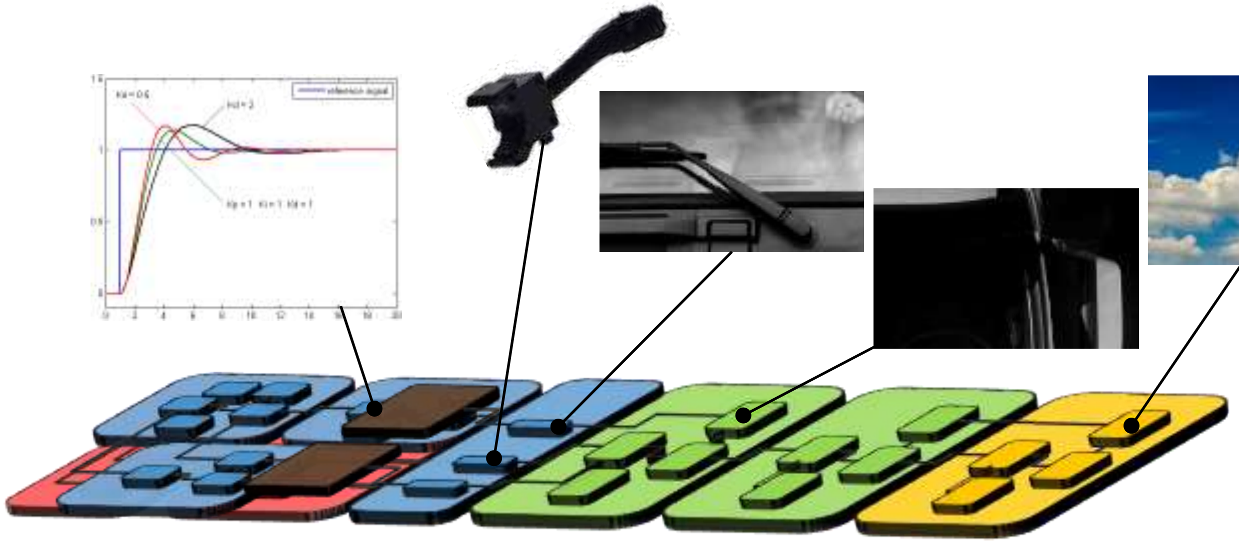
Modelling & Simulation Technology





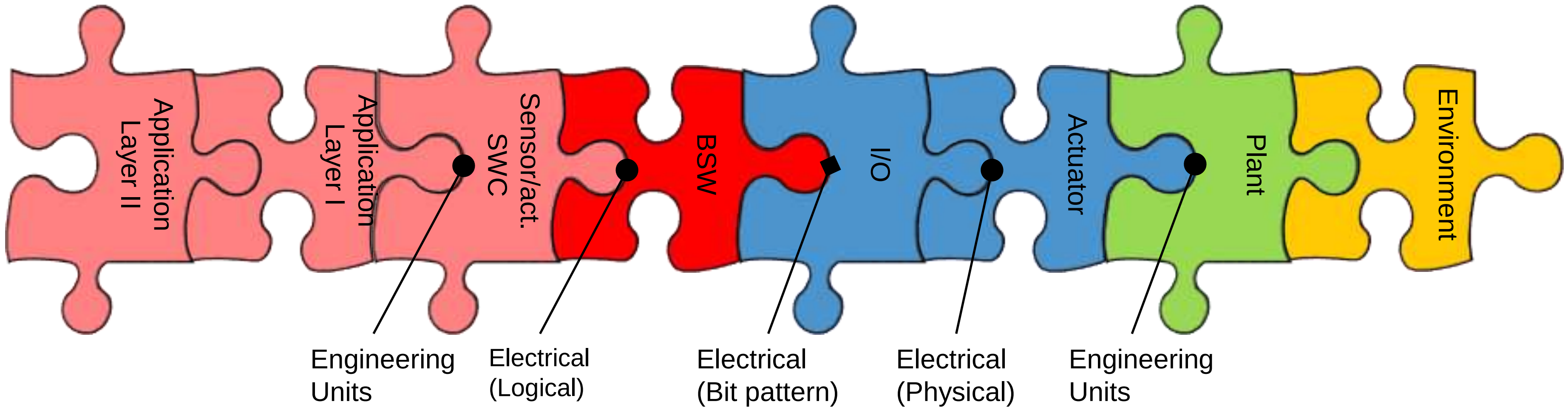
Why
What
How

Modelling Pattern to Support Integration

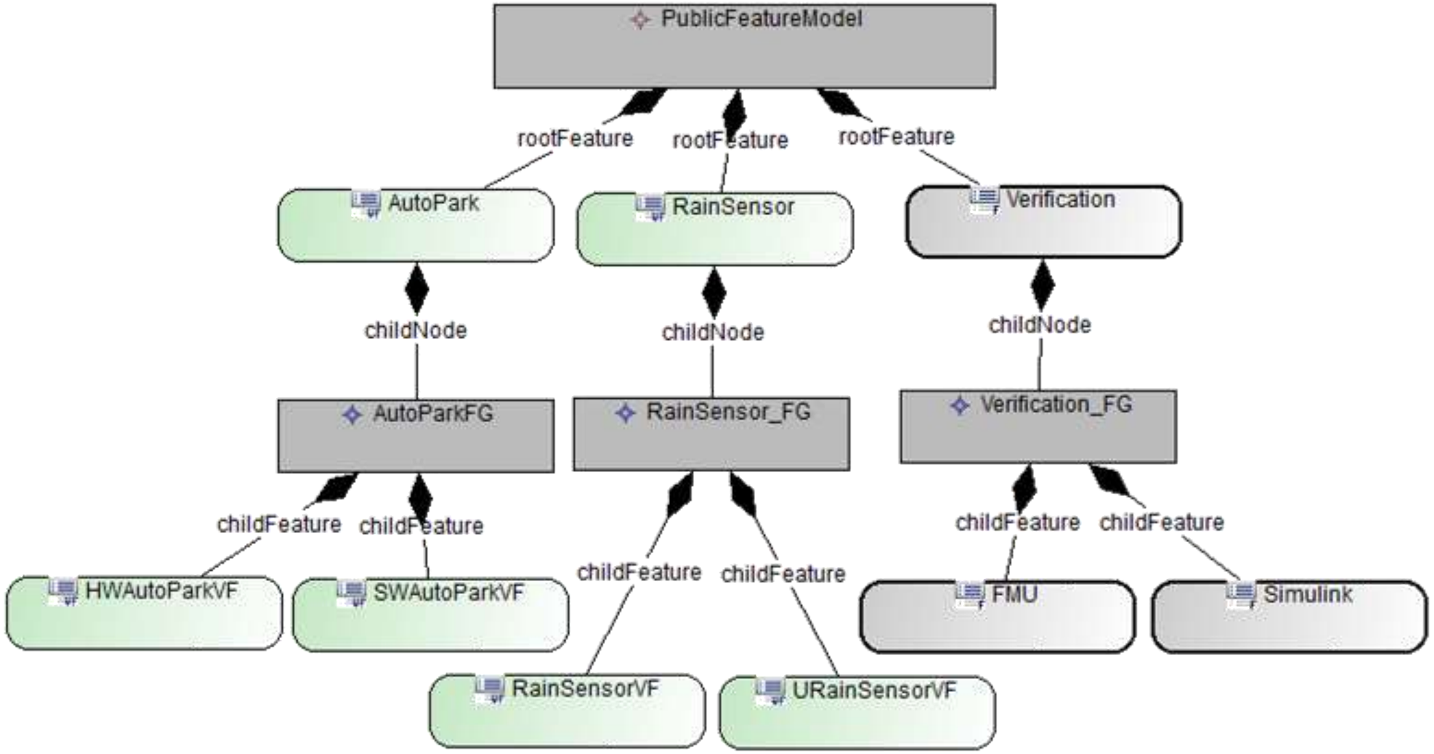


Model-in-the-loop: **Controller model** interfaces to models of sensors and actuators
Software-in-the-loop: **Software** interfaces to models of electronics
Hardware-in-the-loop: **ECU** interfaces to models of sensors and actuators
Hardware⁺-in-the-loop: **Sensor & actuator** interfaces to models of plant

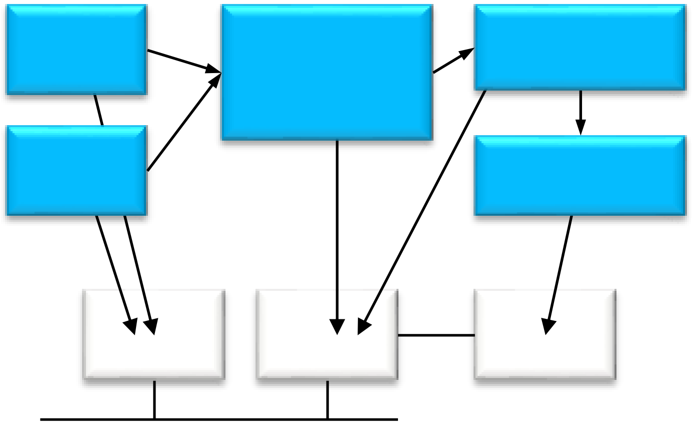
Model Structure



Feature Models



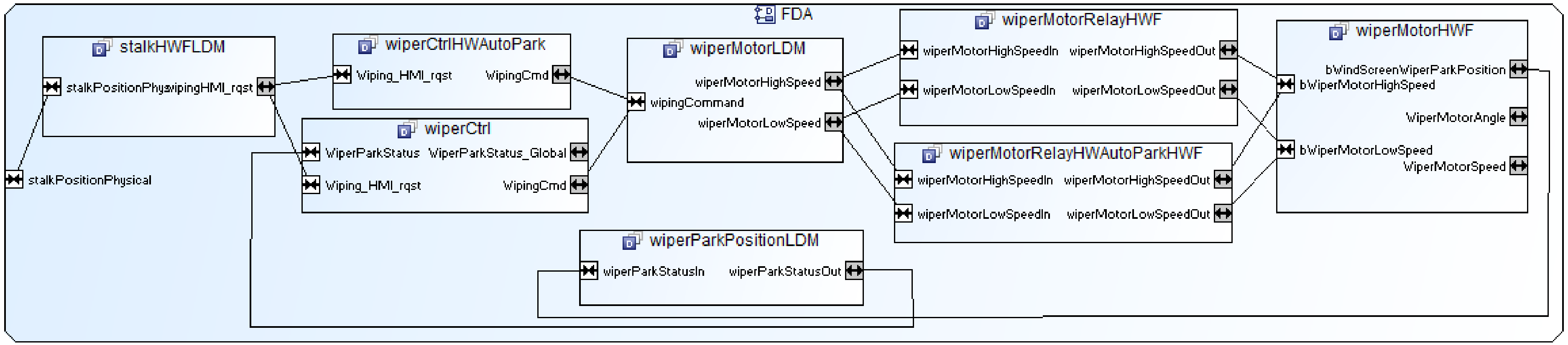
Functional & Software Architecture



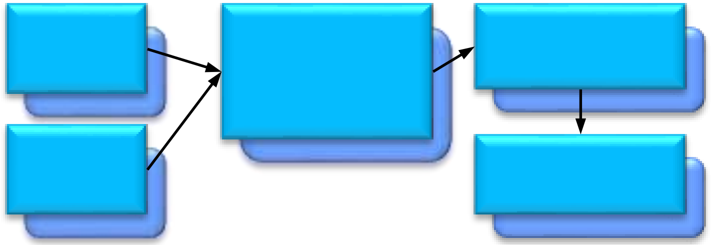
Structure

Function Components
SW Components
HW Topology

EAST-ADL
AUTOSAR

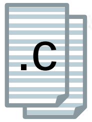


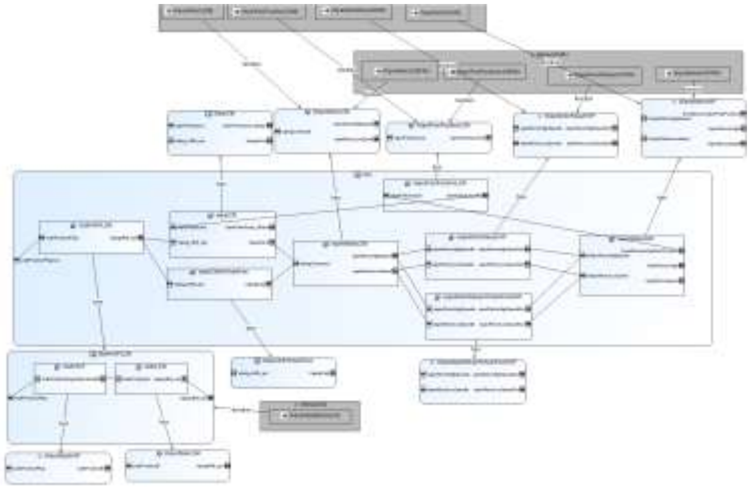
Simulation Content



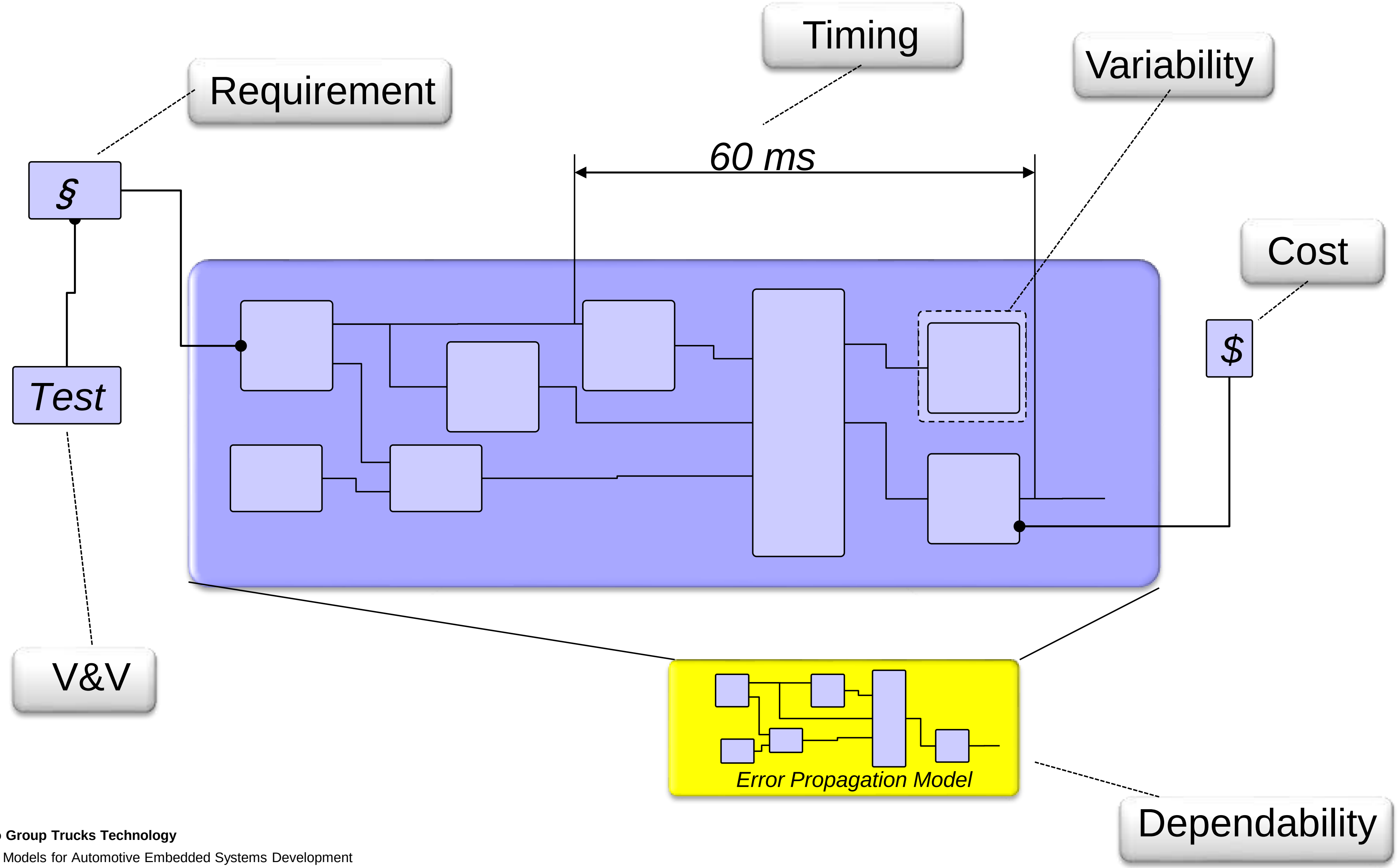
Behavior/Timing

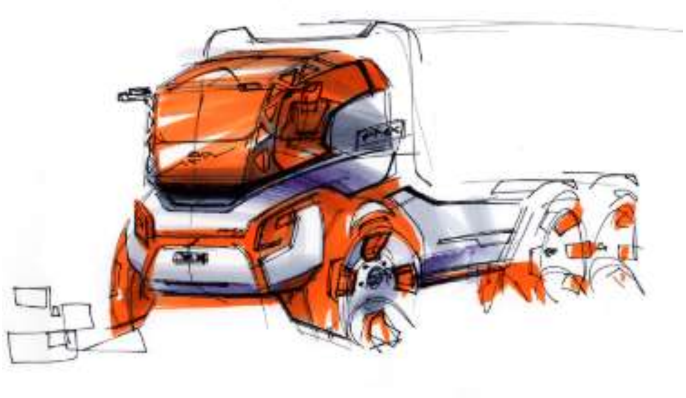
Function Mockup Units+Execution Definition
Source Code+Execution Definition





Property Annotations

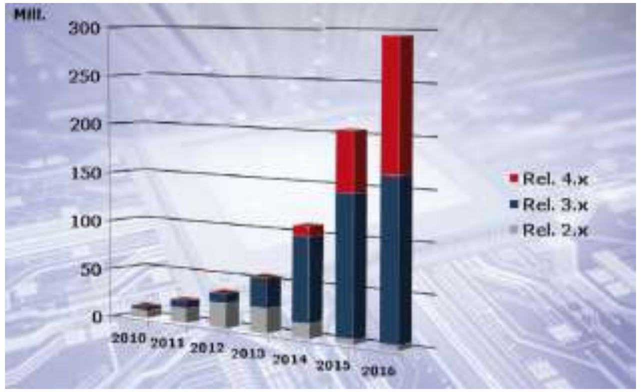




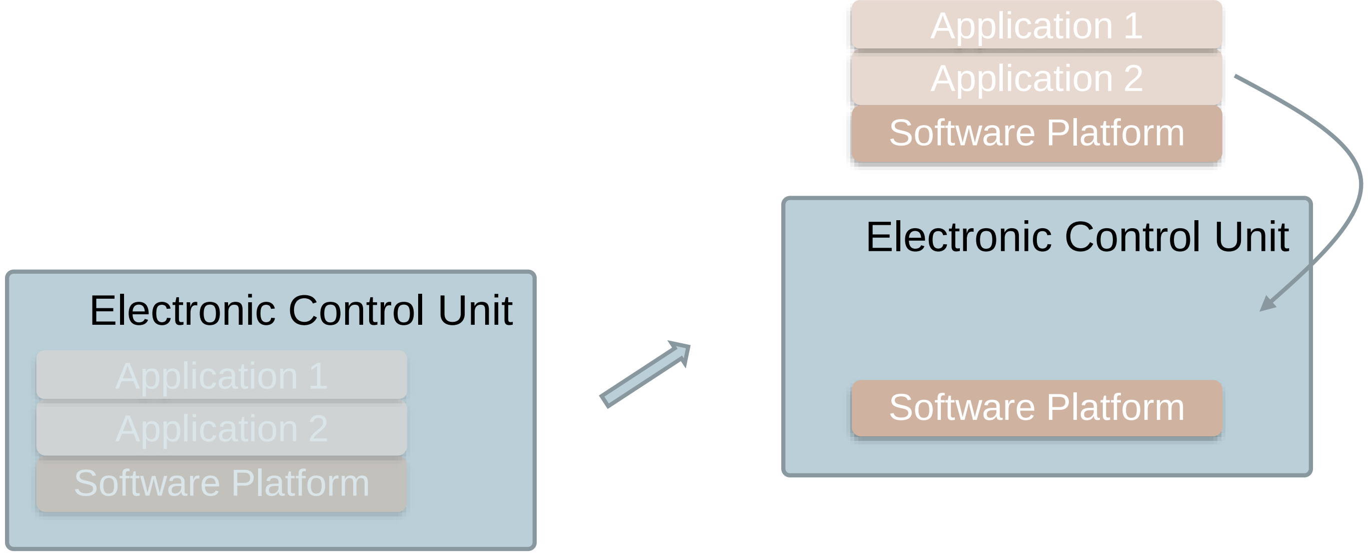
Why
What
How

AUTOSAR

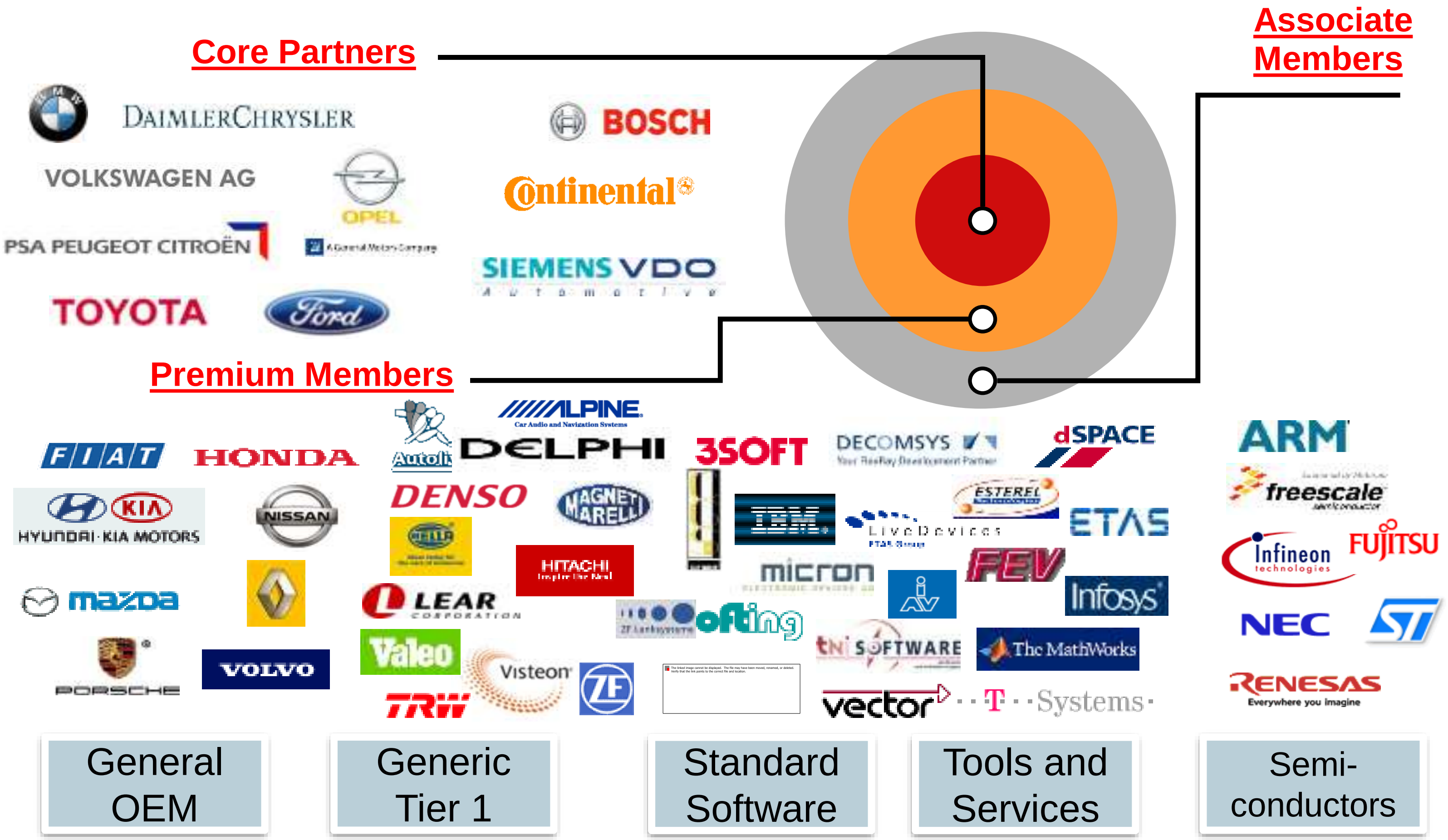
AUTOSAR - Technical Goals



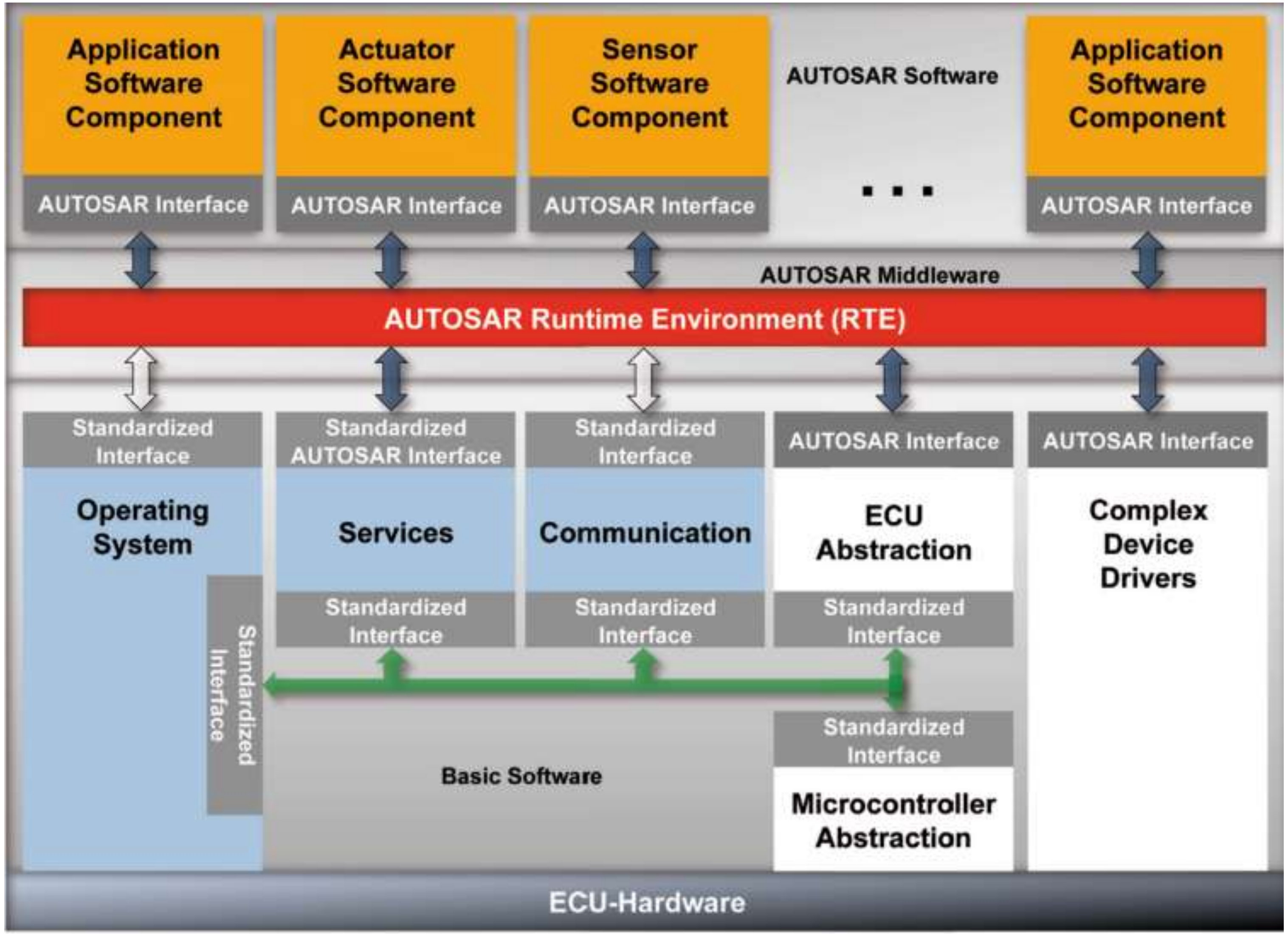
- Increased Flexibility
  - Modularity
  - Scalability
  - Transferability
  - Re-usability
- Standardized platform
  - Off-the-shelf purchase & integration of comm, OS, diagnosis, drivers, etc.
  - Off-the-shelf hardware
- Standardized Interfaces
  - Off-the-shelf purchase & integration of common vehicle functions



AUTOSAR - Consortium

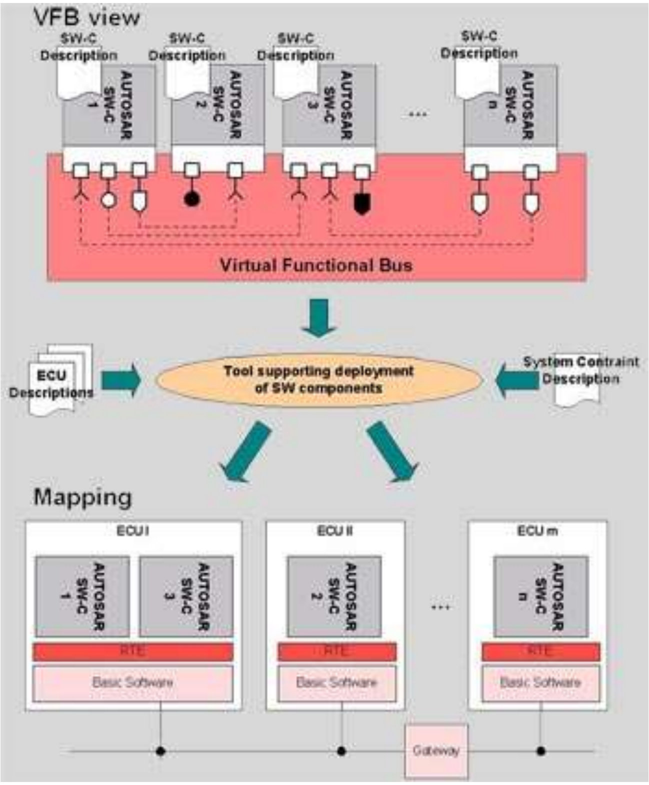


AUTOSAR ECU SW Architecture



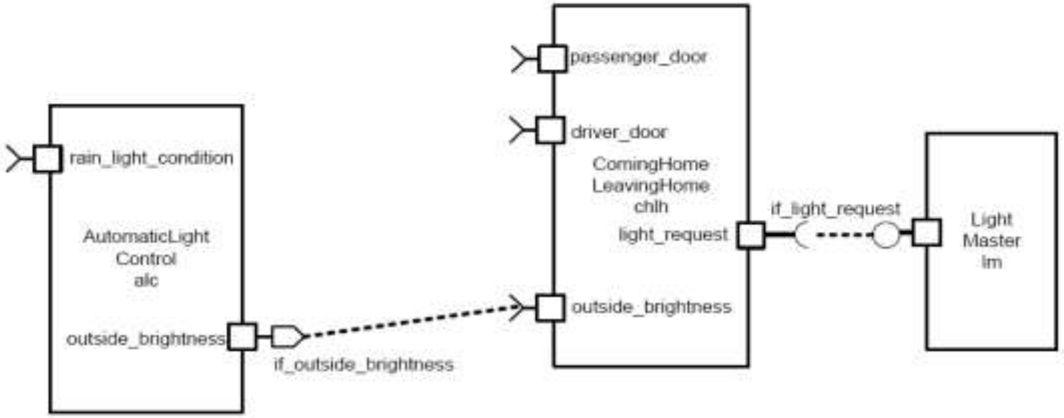
AUTOSAR - Elements

- Modelling
  - Capture SW Components SW Component Template
  - Capture ECU resources: ECU Resource Description
  - Capture allocation and communication: System Description
- Methodology
  - Autogenerate ECU configuration
  - Autogenerate platform SW configuration
  - Autogenerate glue code (RTE)
- Application Interfaces
  - Standard interface definitions for well-established functions
- Architecture
  - Standard platform SW
  - Standard interfaces



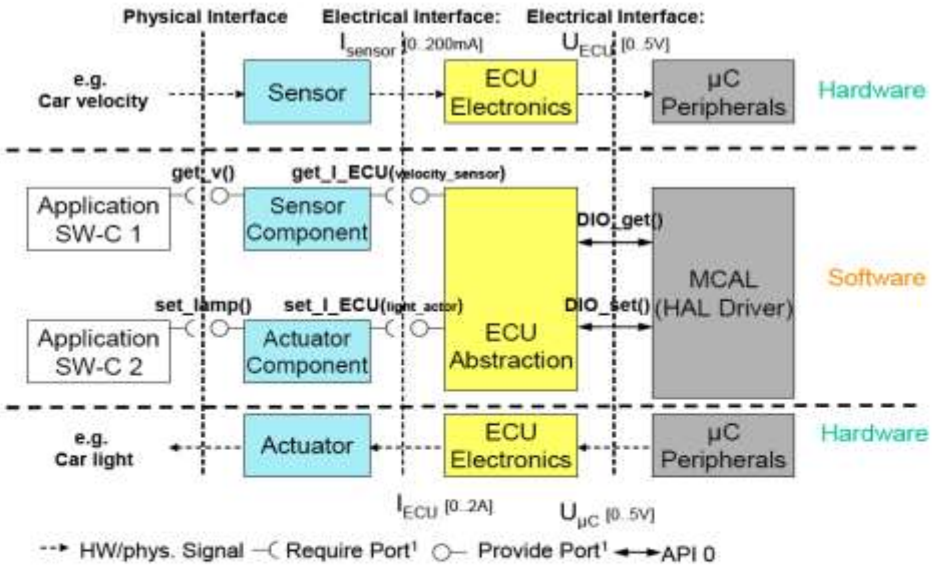
AUTOSAR Software: Software Components

- Hardware independent (CPU and peripherals of ECU)
- Location independent (Communication technology, platform services)
- Specified by
  - Interface
  - Resource Requirements
  - Implementation constraints



AUTOSAR Software: Sensors and Actuators

- SensorActuatorSoftwareComponent adapts device interface to application functions
- Software architecture to separate ECU and device specific parts
- Support for re- allocation of sensors and actuators



AUTOSAR Basic Software

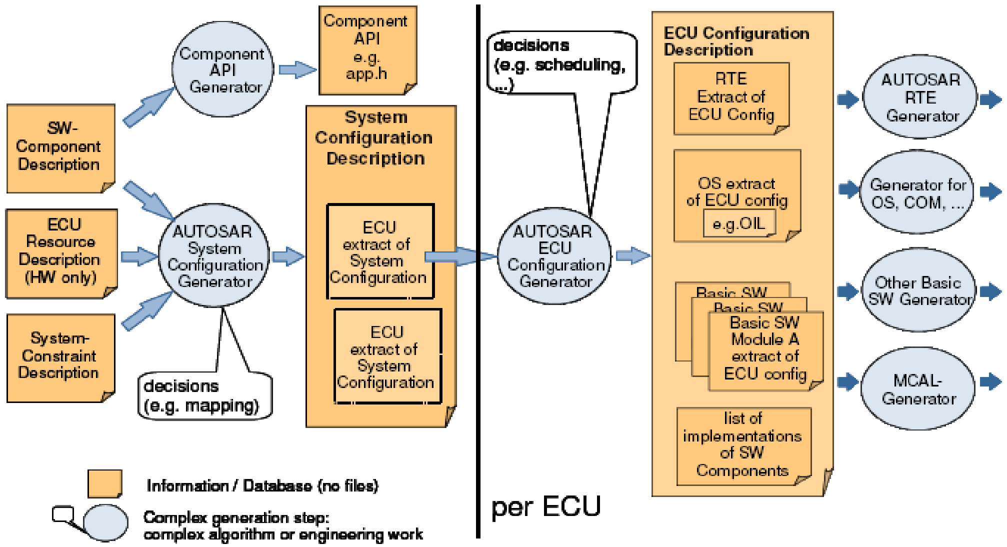
- **Software platform for application SW components**
- **Standardized Components**
  - System services (diagnostic protocols; NVRAM, flash and memory management)
  - Communication (CAN/LIN/FlexRay... framework, I/O management, Network management)
  - Operating System (OSEK based)
  - Microcontroller Abstraction
- **ECU Specific Components**
  - ECU Abstraction
  - Complex Device Driver

AUTOSAR Methodology

Covers process steps for

– ECU platform software generation and configuration

– Application software intregration

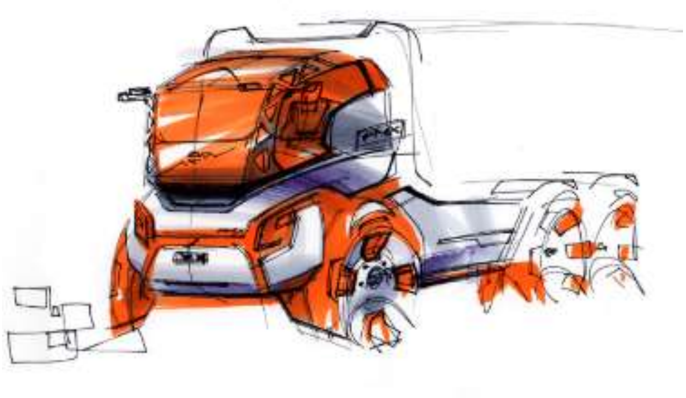


Why

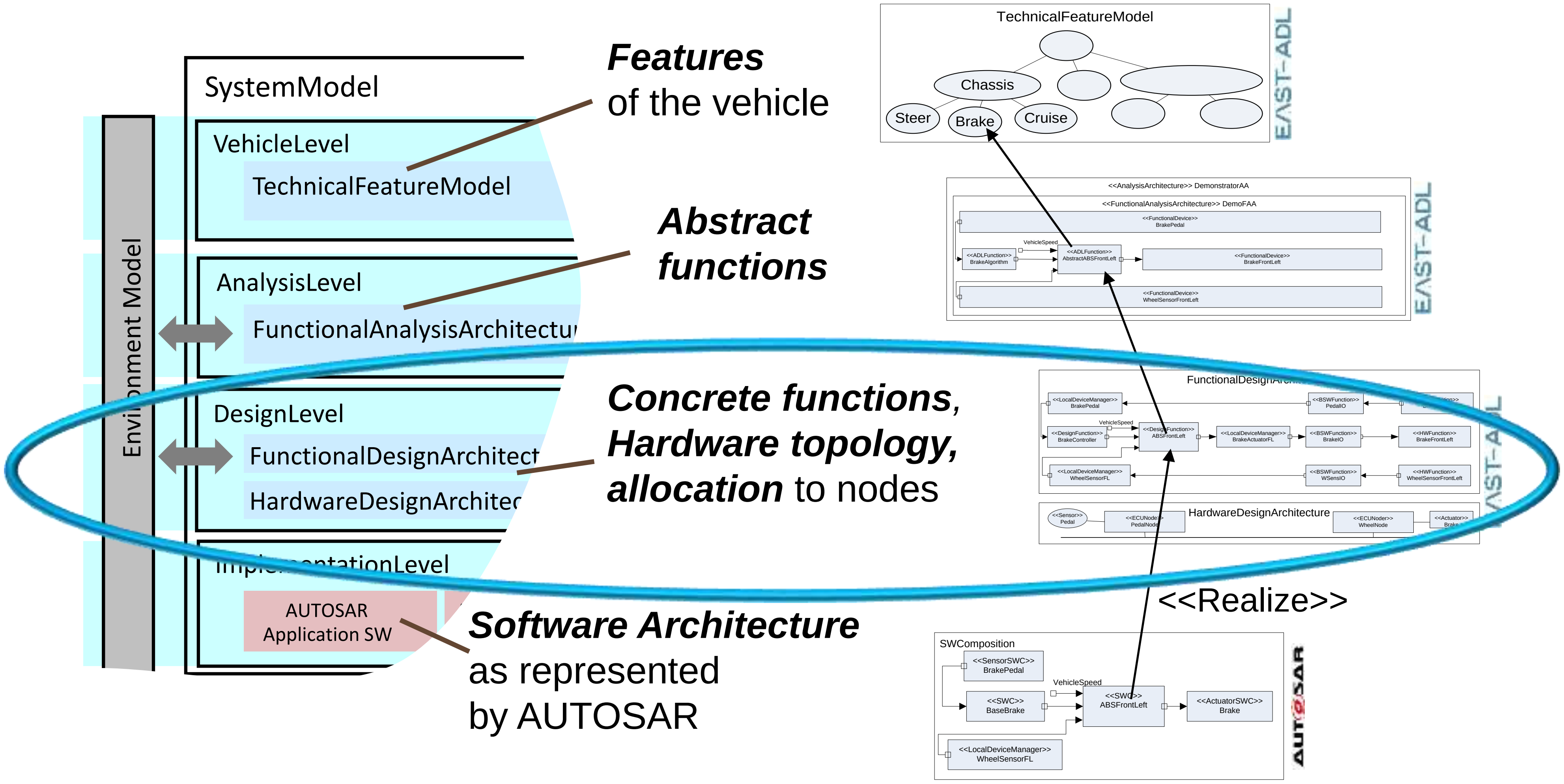
What

How

EAST-ADL

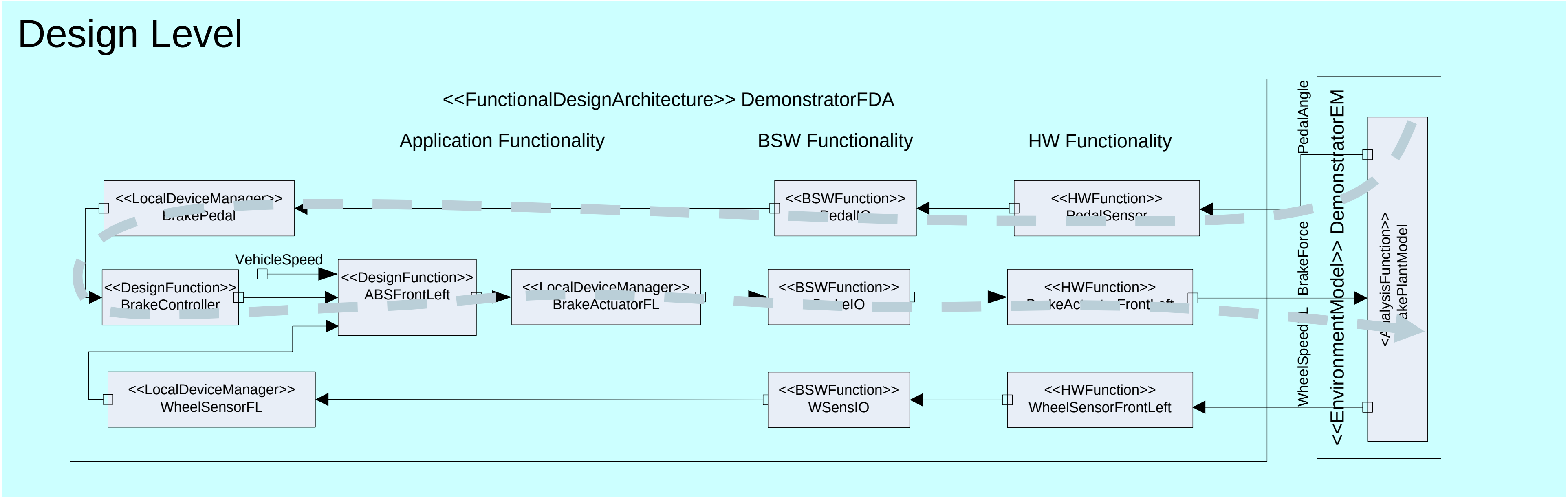


EAST-ADL Representation



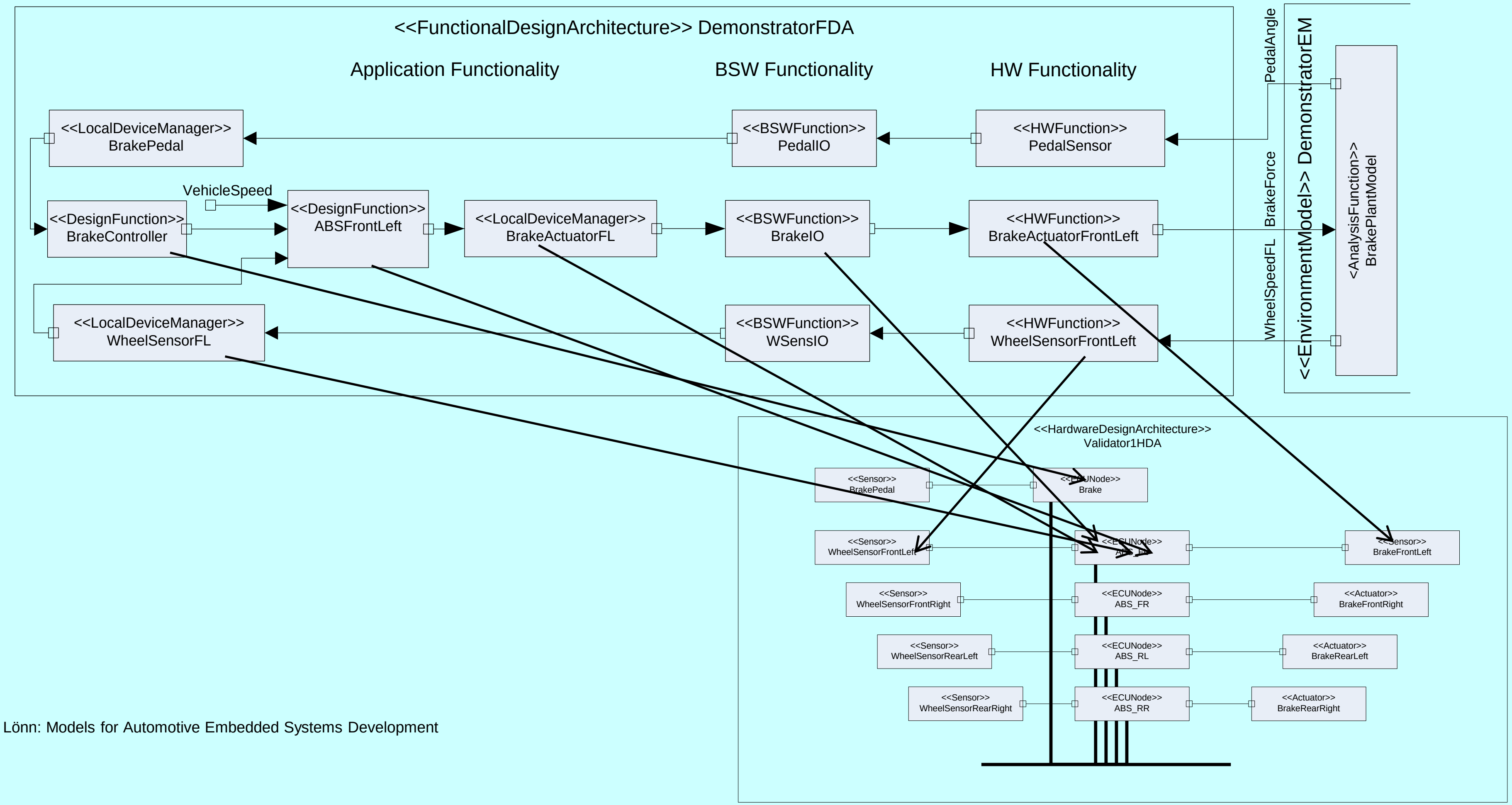
EAST-ADL Model on Design Level

Functional Structure



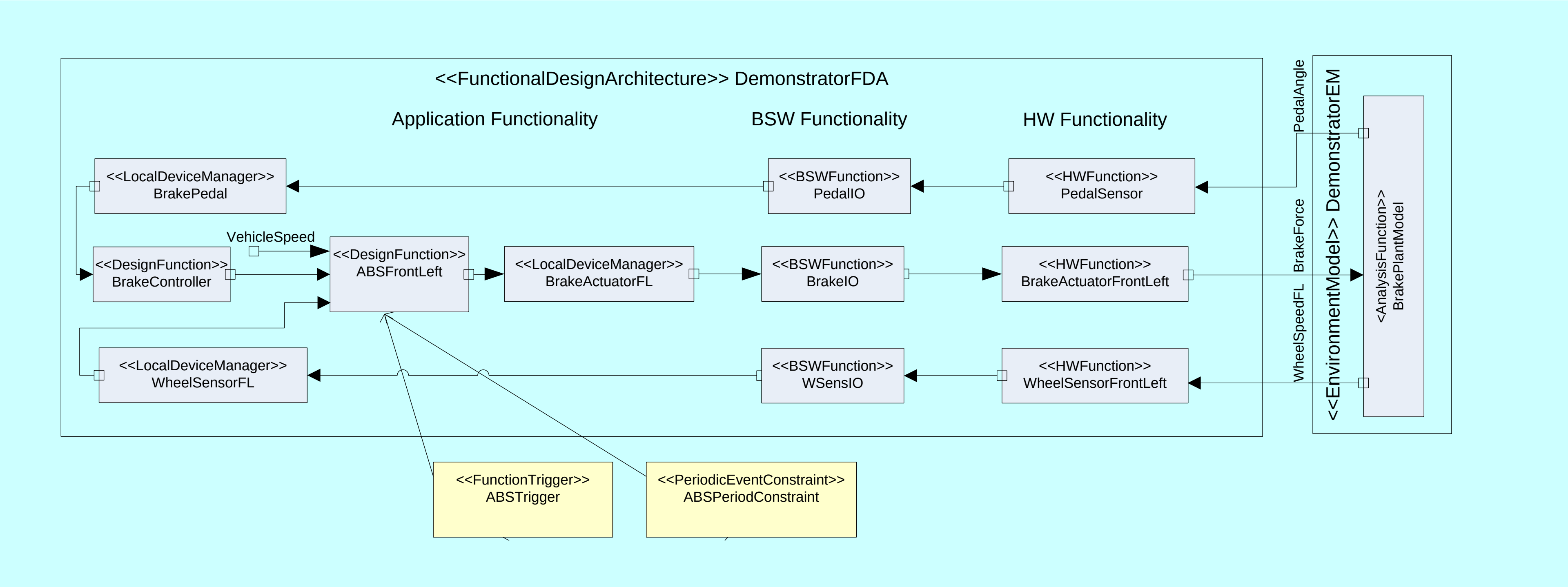
EAST-ADL Model on Design Level

Functional Structure onto HW Architecture



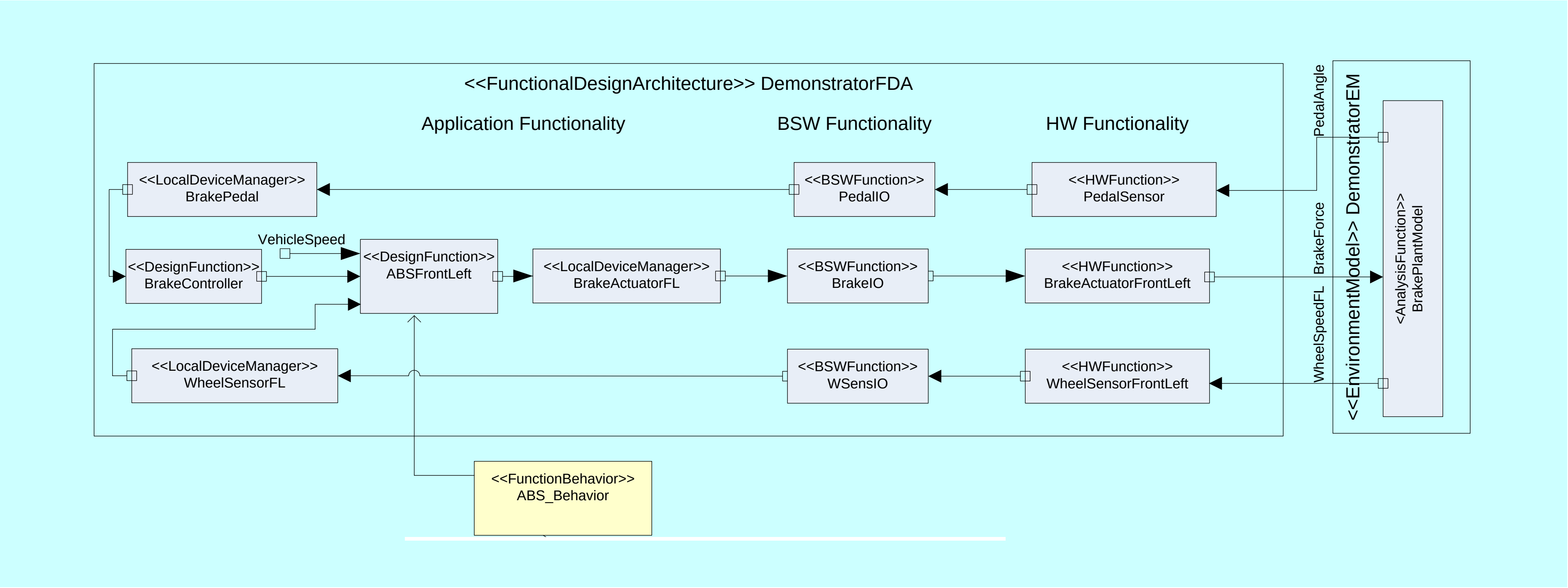
EAST-ADL Model on Design Level

Timing/Triggering



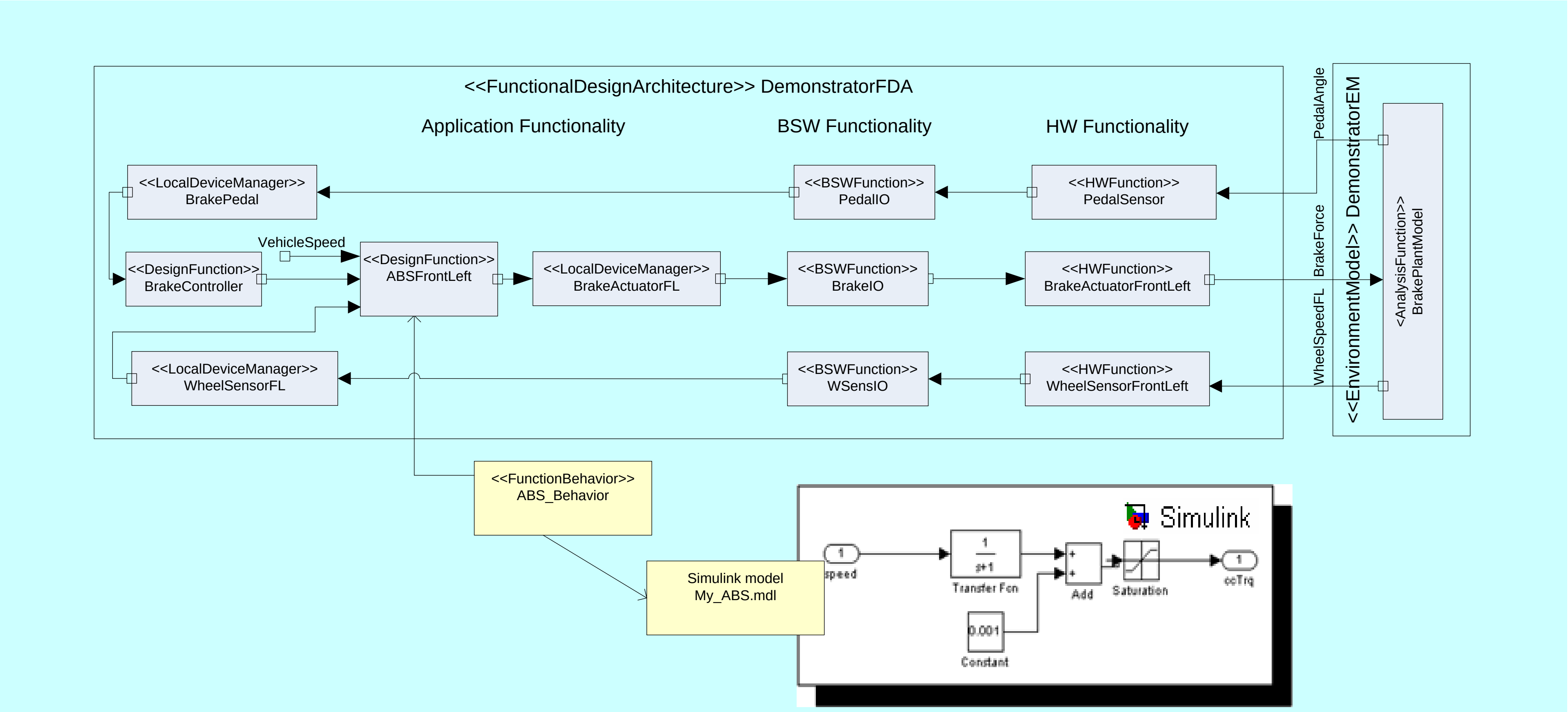
EAST-ADL Model on Design Level

Transfer Function – ”Black-box” behavior



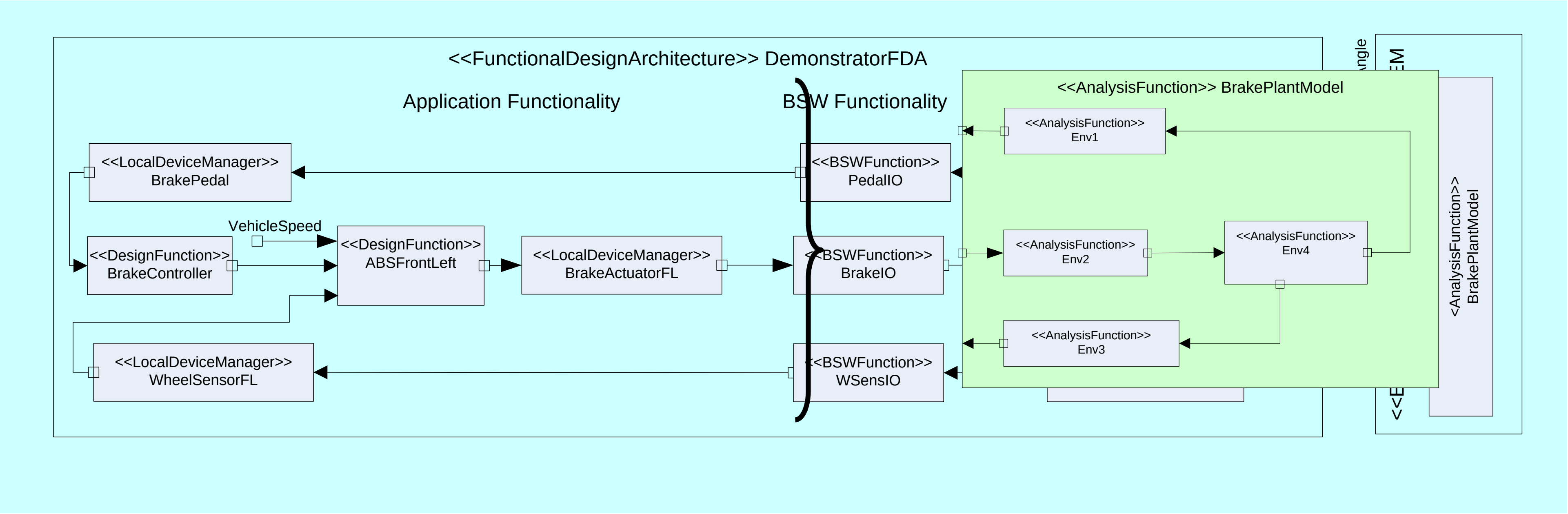
EAST-ADL Model on Design Level

Transfer Function – ”Black-box” behavior



EAST-ADL Model on Design Level

Behavior of environment (Plant)



Typically in plant model:

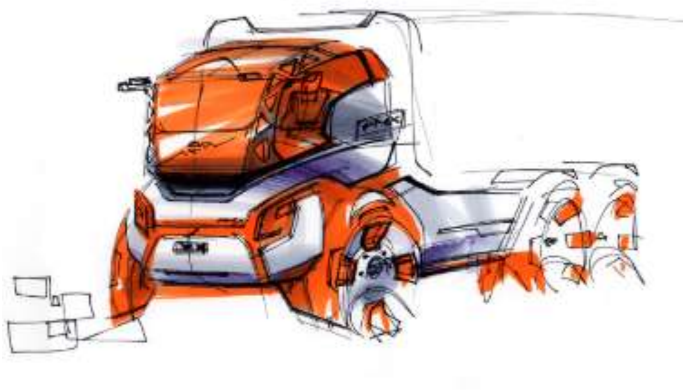
- Non-causal: ”Power Ports” – {Torque,Speed}, {Pressure, Flow}, etc.
- Continuous Time: No triggers

Why

What

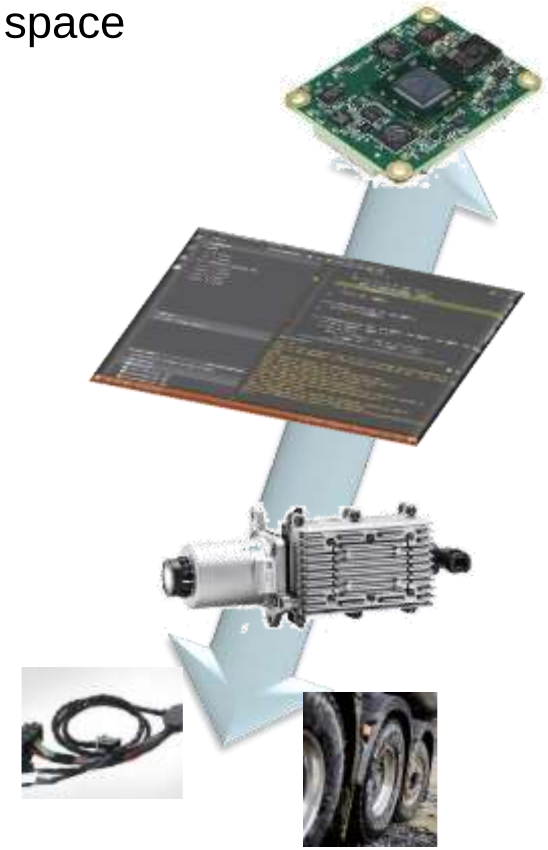
How

Variability & Product Lines

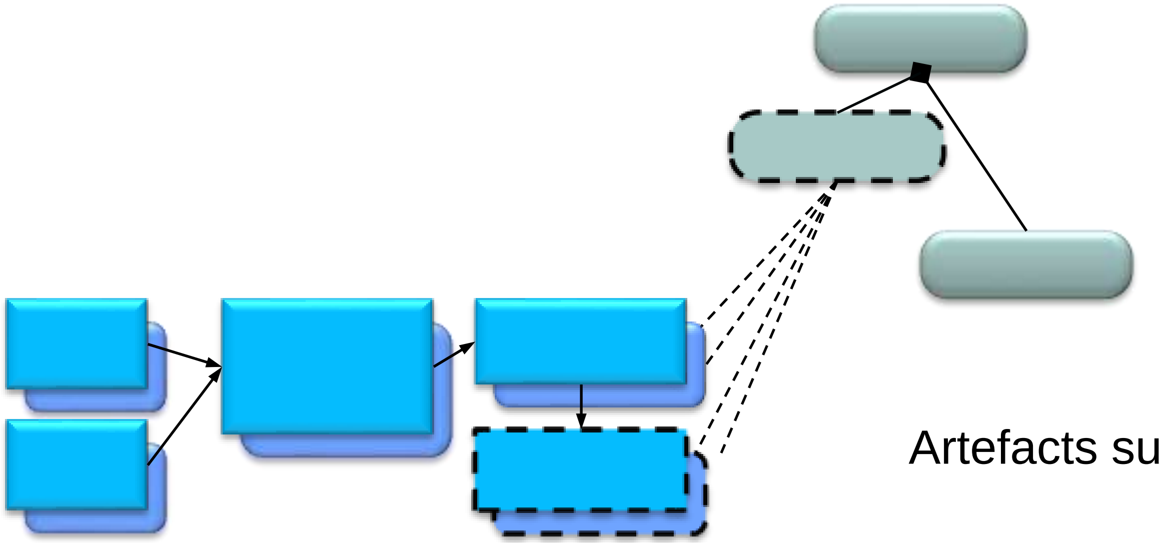


Variability - Background

- On average, less than 2 vehicles of the same configuration is built: Huge configuration space
- All engineering information is governed by configuration and variant
  - Requirements
  - Components
  - Tests
  - Annotations
  - ...
- Co-variation SW and mechanics/plant
  - SW configuration must be consistent with mechatronics and mechanics of truck
- Customer variability vs technical variability
  - Some variability is invisible to the user: Internal technical solutions
- Detaled variability vs. Exposed variability
  - Some details are only relevant for the component developer



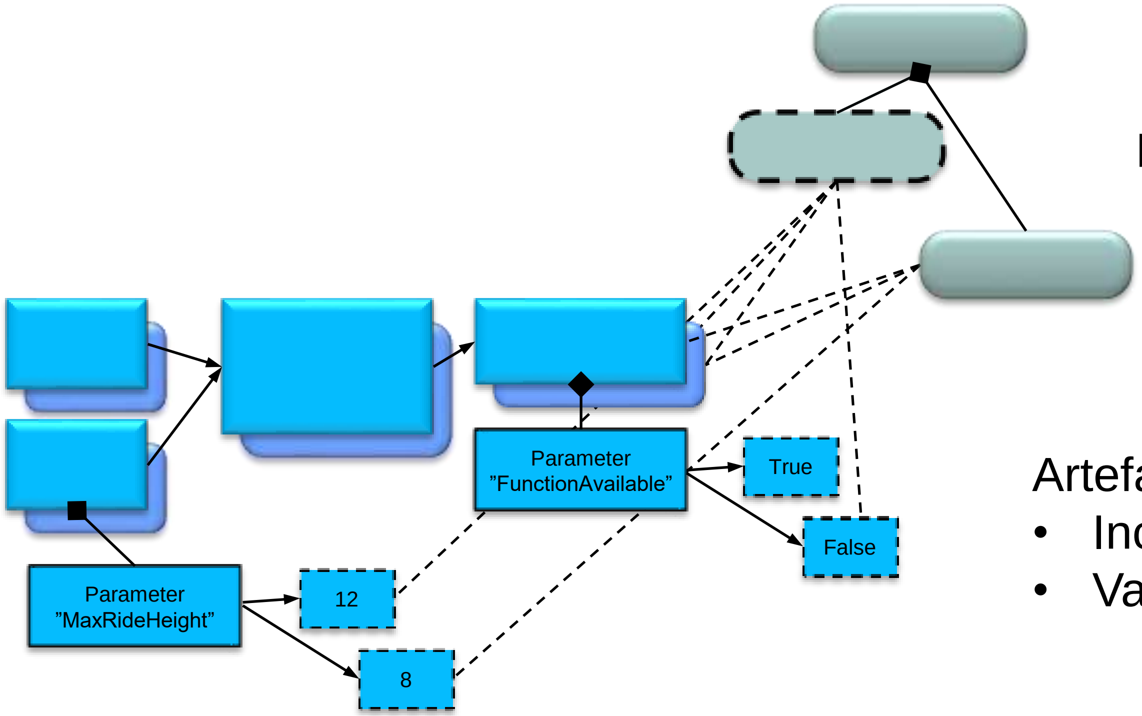
Variability Mechanism 1: Include/exclude



Feature model for configuration

Artefacts subject to configuration

Variability Mechanism 2: Parameterize

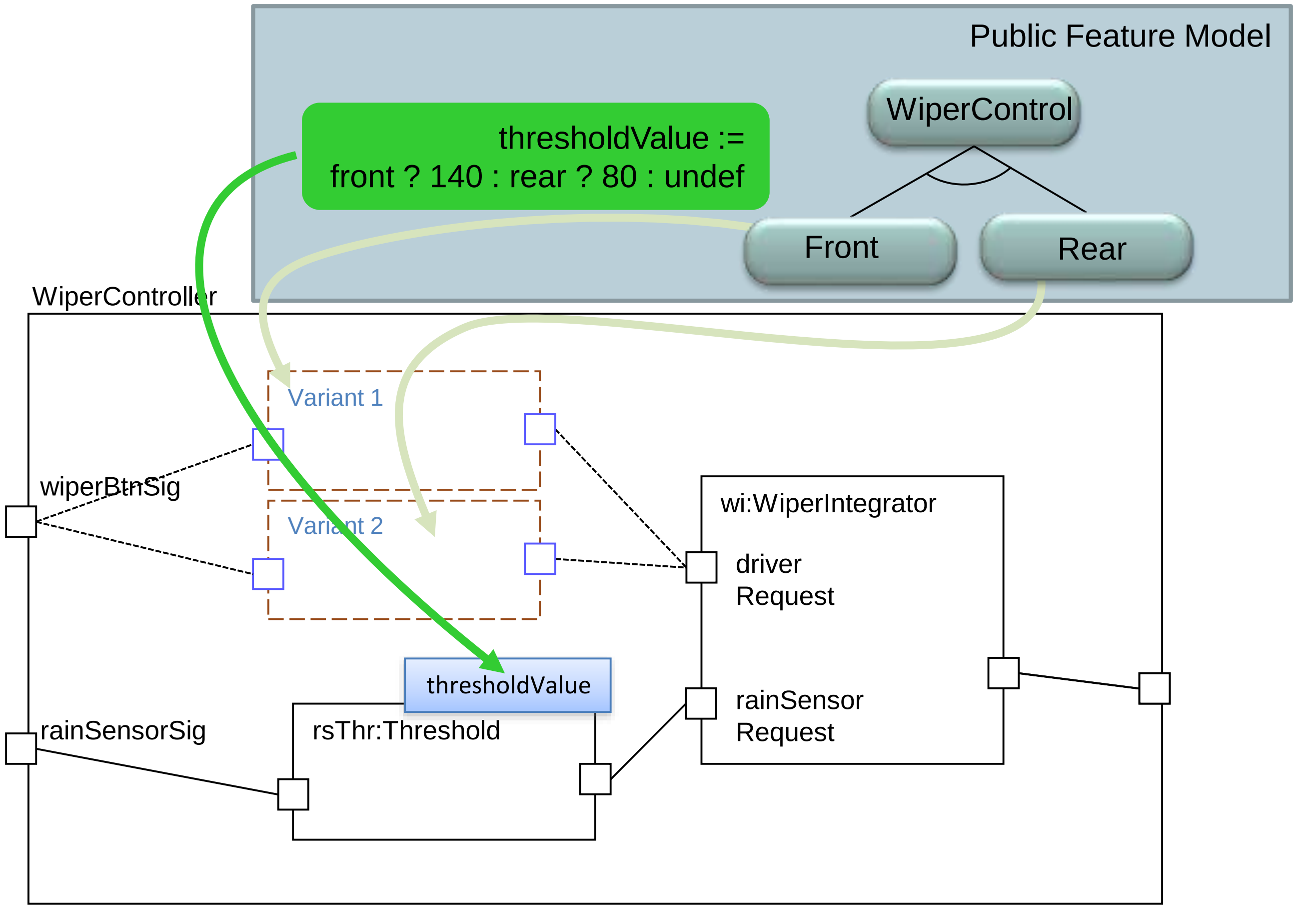


Feature model for configuration

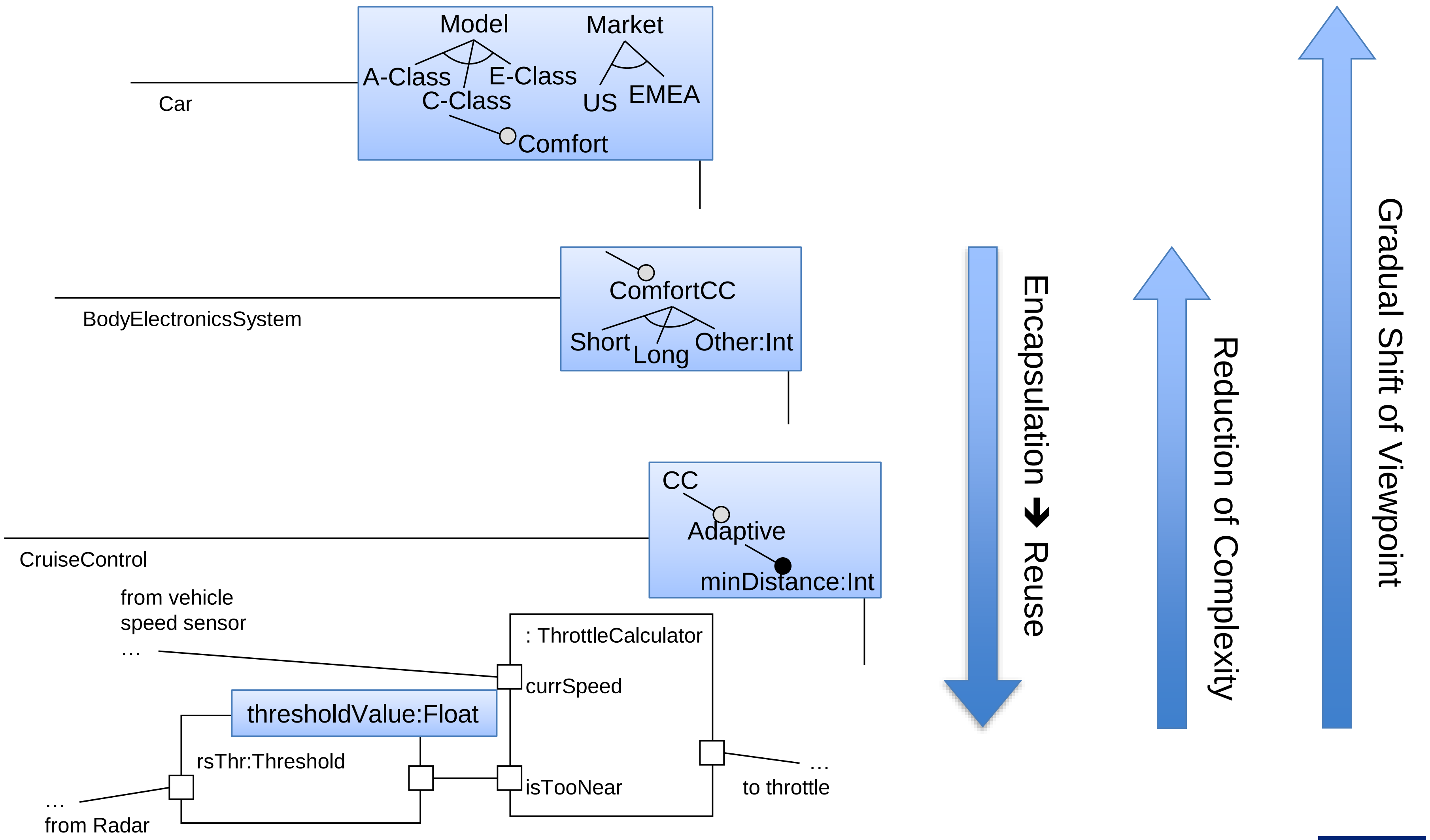
Artefacts subject to configuration

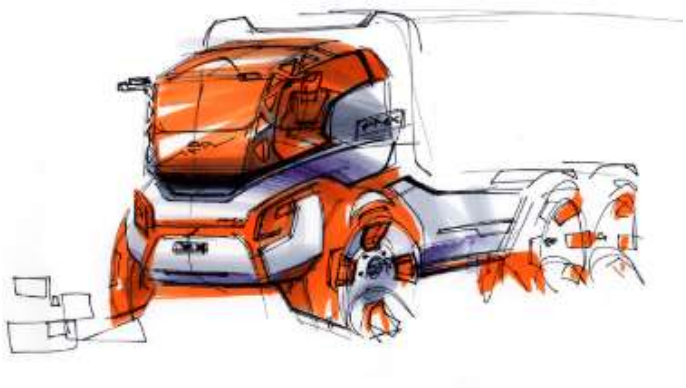
- Include/exclude
- Variant-specific calibration

Compositional Variability



Compositional Variability





Why

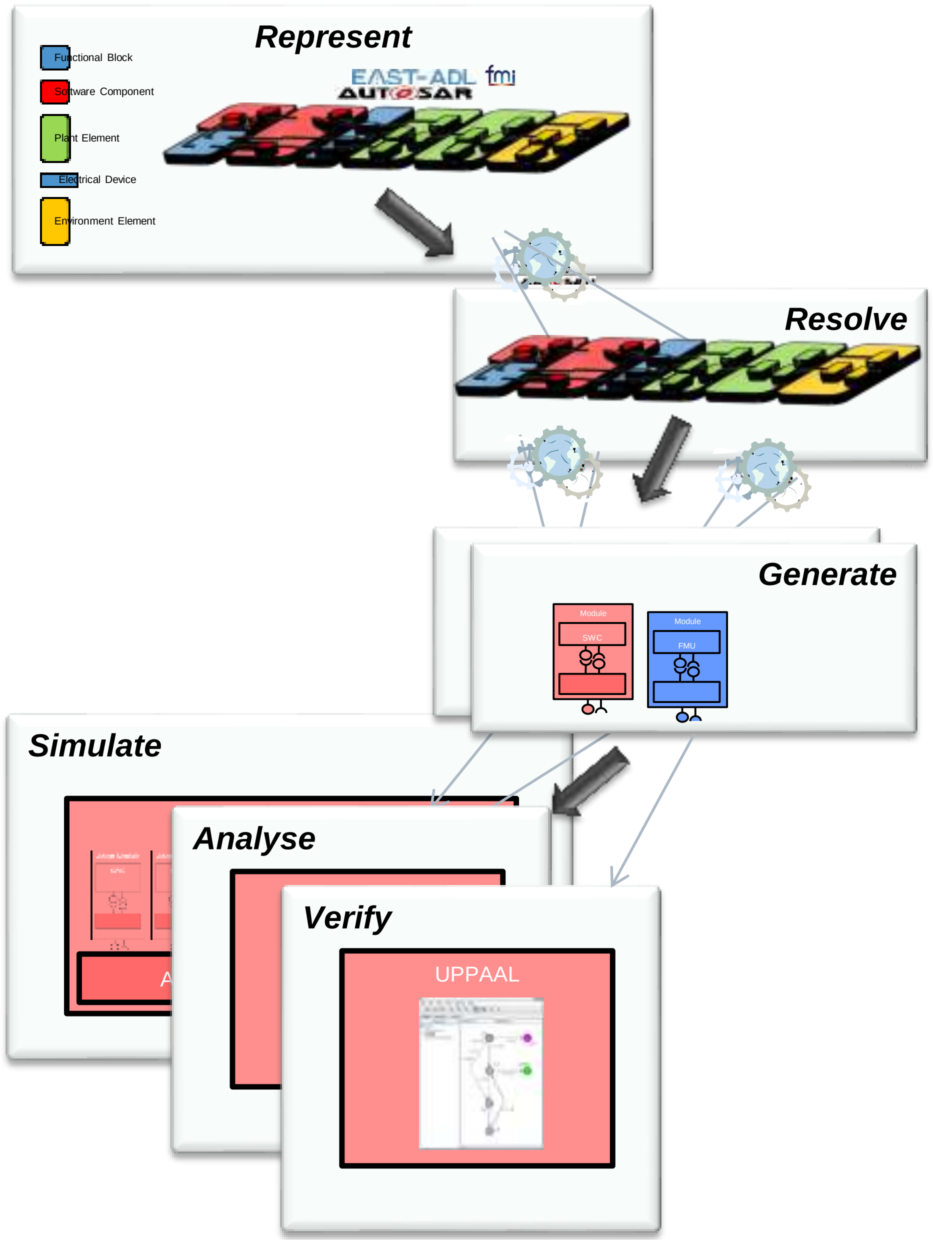
What

How

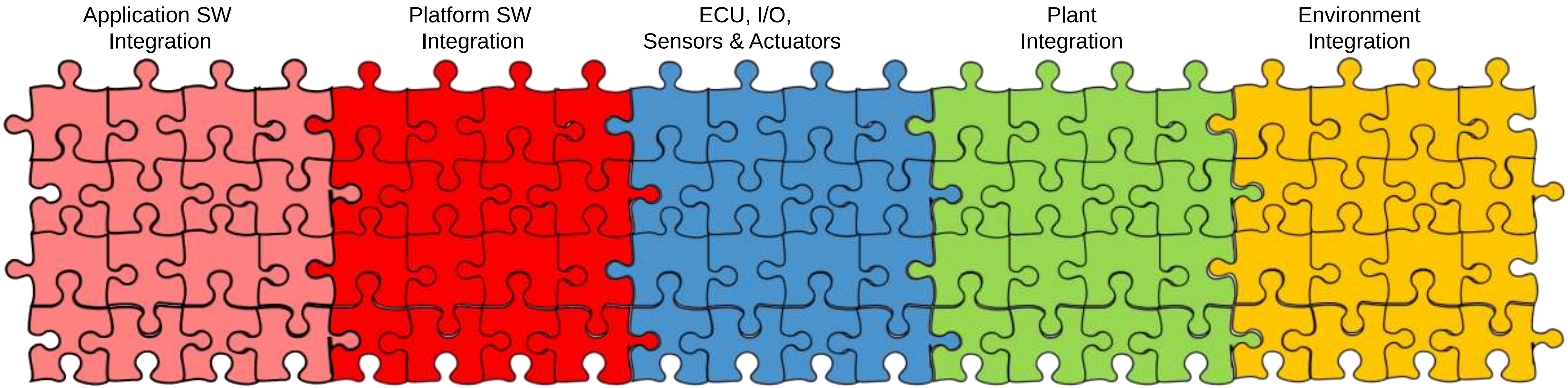
Workflow

Engineering Workflow

- Represent
  - Models and Code
- Resolve
  - Configure and Resolve variability
- Generate
  - Automatic preparation for each tool
- Simulate/Analyze/Verify
  - Software centric simulations
  - Physics centric simulations
  - Arithmetic analysis
  - Formal verification
  - ...



Configuration



Configuration

Software:

AUTOSAR SWC Selection

AUTOSAR SWC Parameterization

- Configuration ports
- Initialization values

Non-Software

Logical Component Selection

Function Mockup Unit Parameterization

- Configuration Ports
- Initialization values

Environment:

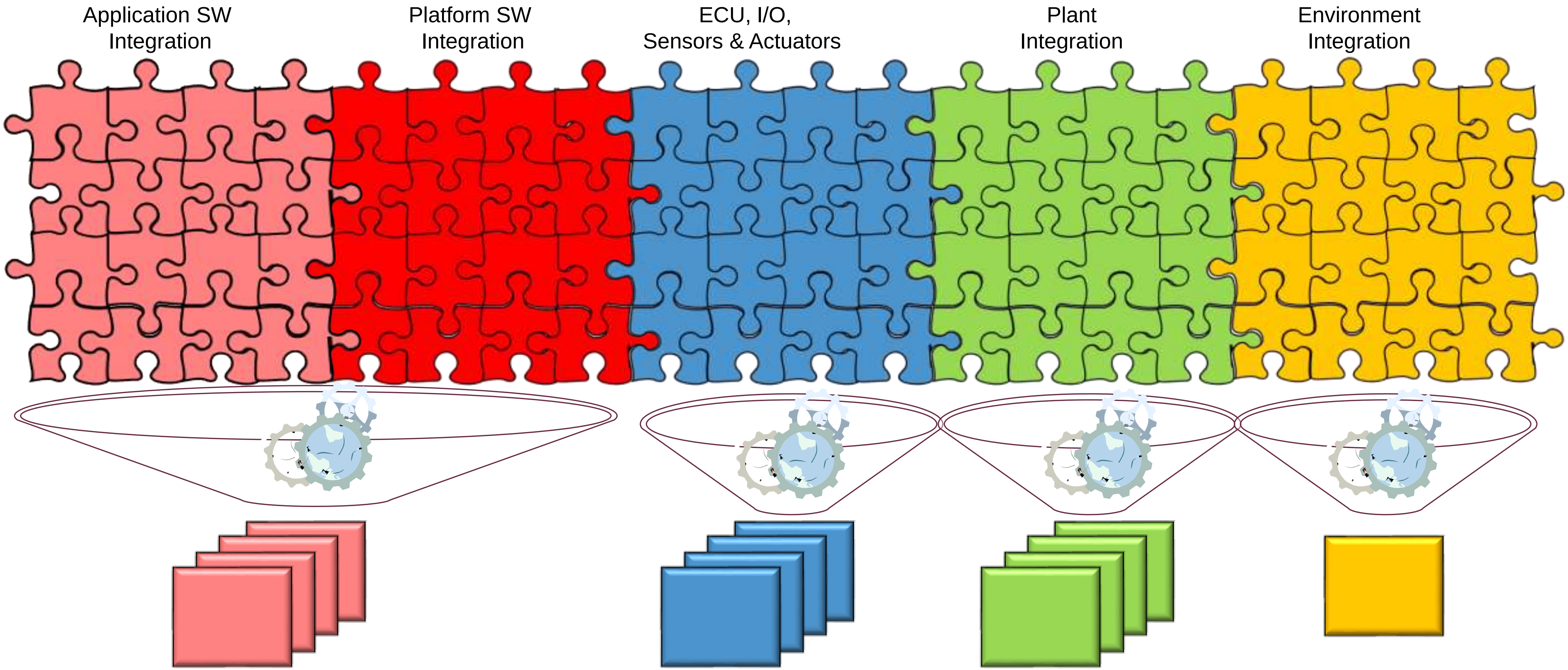
Traffic Scenario:

- OpenScenario Selection & Parameterization

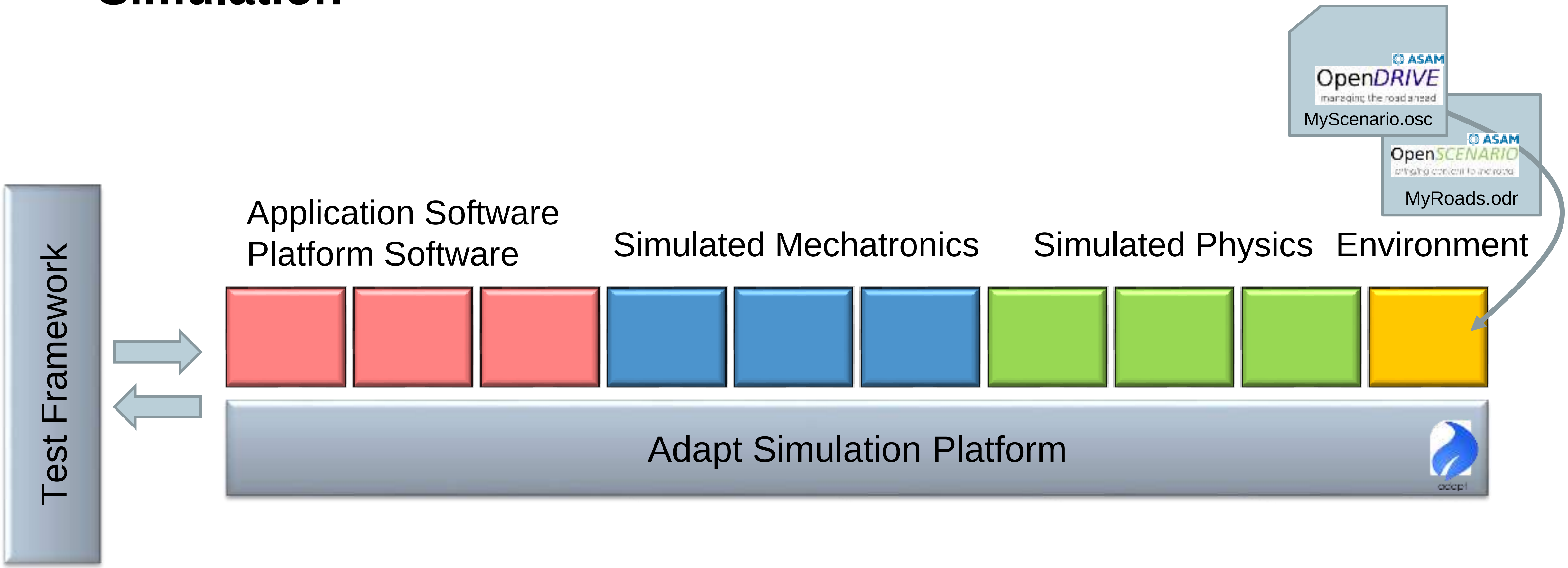
Roads:

- OpenRoad Selection and Parameterization

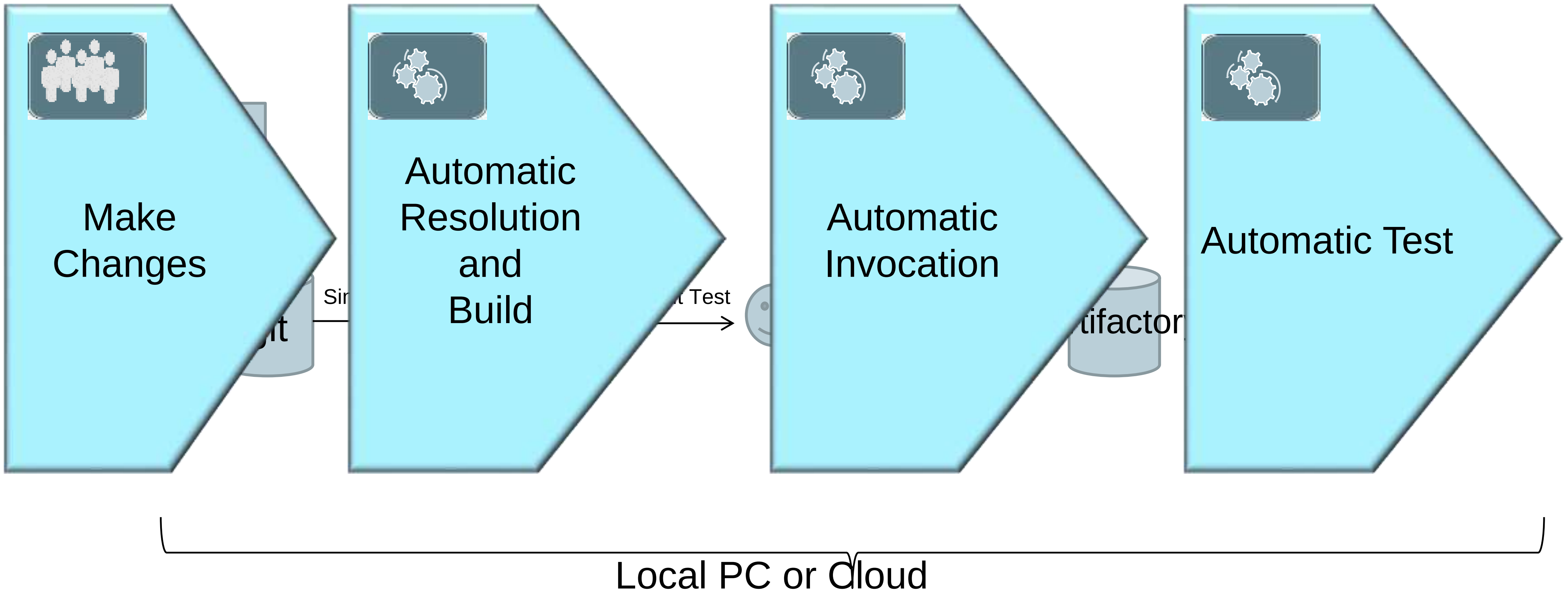
Generation


Application SW
Integration
Platform SW
Integration
ECU, I/O,
Sensors & Actuators
Plant
Integration
Environment
Integration

Simulation


MyScenario.osc
MyRoads.odr
Application Software
Platform Software
Simulated Mechatronics
Simulated Physics
Environment
Test Framework
Adapt Simulation Platform

Engineering Automation



Conclusion

- AUTOSAR software platform is automotive de-facto standard
- EAST-ADL supports automotive embedded systems modelling
  “starting” with needs and requirements and
  “ending” with an AUTOSAR SW architecture

- An agreed modelling language makes it possible
  – to understand engineering information from other departments/disciplines/companies
  – to exchange engineering models between different organizations and tools
  – to progress jointly on tools and methodology for modelling, analysis and synthesis

Summary

New Vehicle Functionality

Development Efficiency

Short Lead Time

Iterative & Incremental Development

Engineering Automation

Automatic Configurability

Automatic Evaluation

Variability Modelling

System & Plant Modelling