



# EAST-ADL2 Overview

---

Henrik Lönn, Volvo Technology AB  
*CPS Week 2010, KTH, Stockholm*

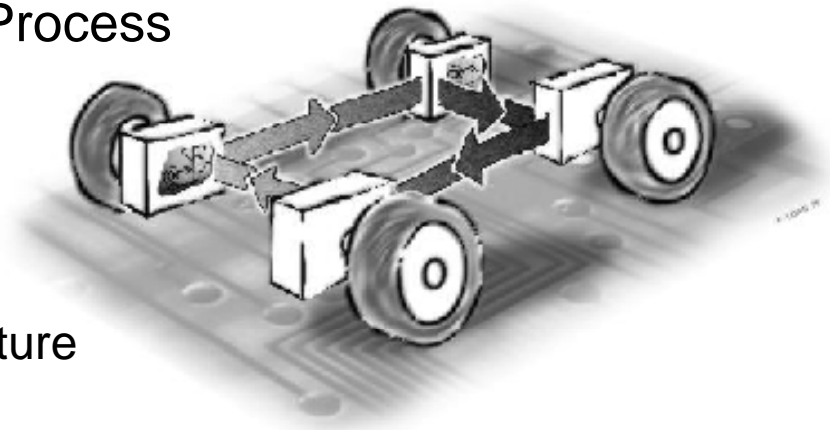
# The Automotive Challenge

## Product Related Challenges

- Functionality increase
- Complexity increase
- Increased Safety-criticality
- Quality concerns

## Challenges Related to Development Process

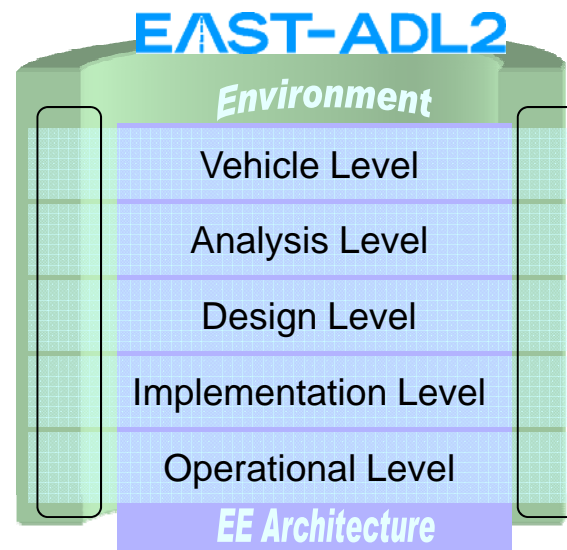
- Supplier-OEM relationship
- Multiple sites & departments
- Product families
- Componentization
- Separation of application from infrastructure
- Safety Requirements, ISO 26262



# Response: EAST-ADL2

A System Modeling Approach/Architectural Framework that

- Is a template for how engineering information is organized and represented
- Provides separation of concerns
- Embrace the de-facto standard representation of automotive software – AUTOSAR



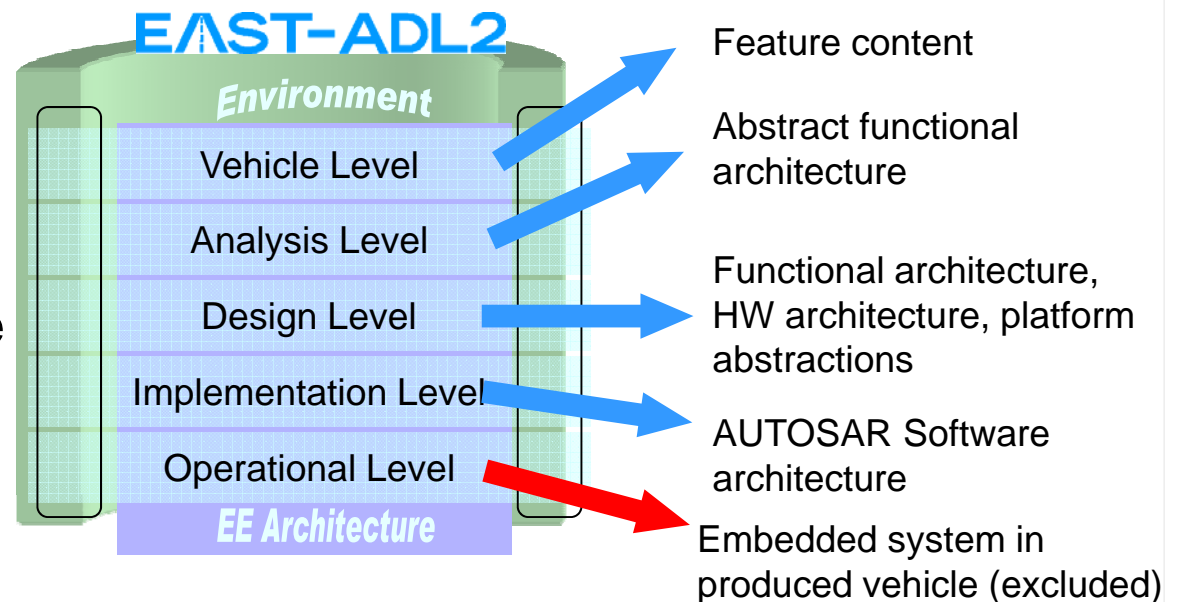
## How is an EAST-ADL2 model structured?

An EAST-ADL2 model is organized in several levels of abstraction, where the software and electronics based artifacts are modeled

The abstraction levels are “views” on the model and a complete representation of the system

The contents on an abstraction level forms a complete representation of the vehicle embedded system, with respect to the concerns of that level

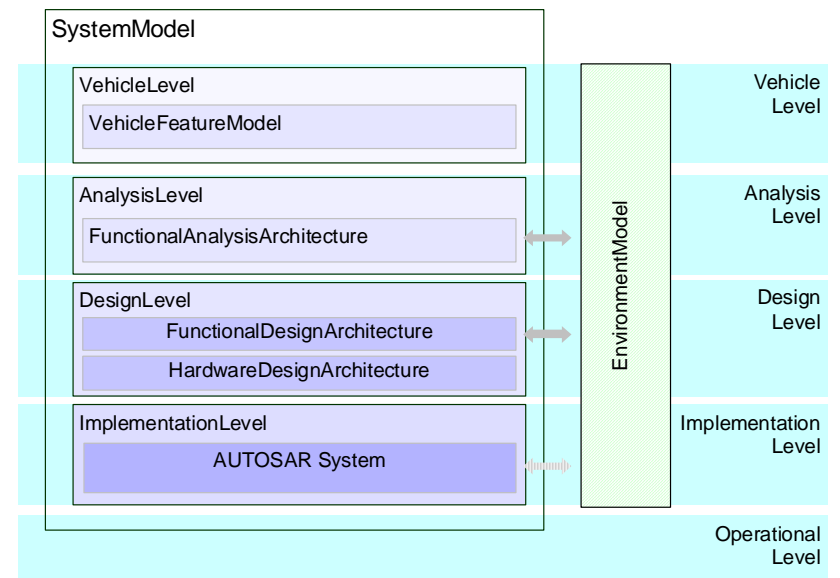
The levels are refined top-down



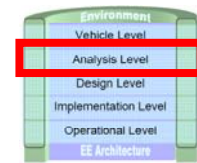
# Vehicle Level



- A Vehicle is characterized by a set of Features
- Features are *stakeholder* requested functional or non-functional characteristics of a vehicle
- A Feature describes "what", but shall not fix the "how"
- A Feature is specified by requirements and use cases
- From a top-down architecture approach the features are the configuration points to create a vehicle variant

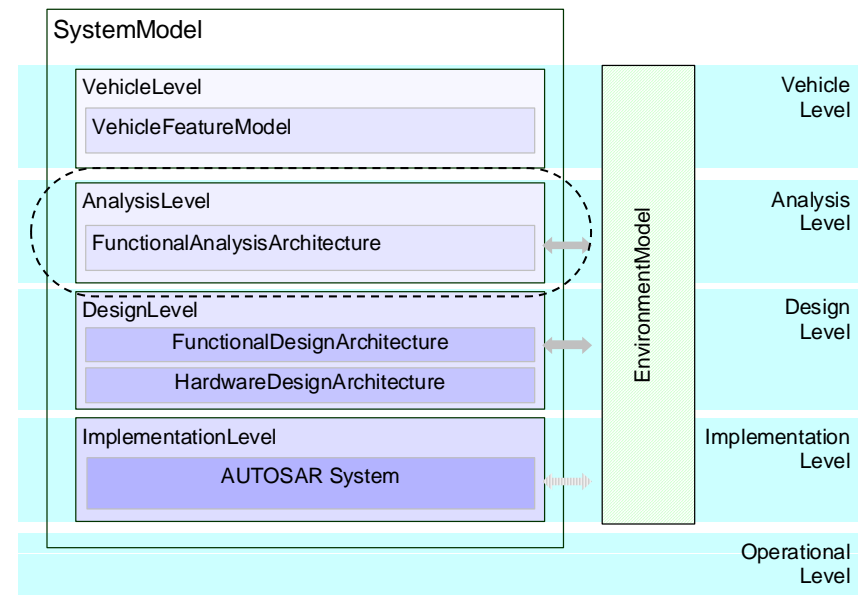


# Analysis Level



Analysis Level is the abstract Functional description of the EE system

- Realizes functionality based on the features and requirements
- Captures abstract functional definition while avoiding implementation details
- Defines the system boundary
- Environment model and stakeholders define context
- Basis for safety analysis

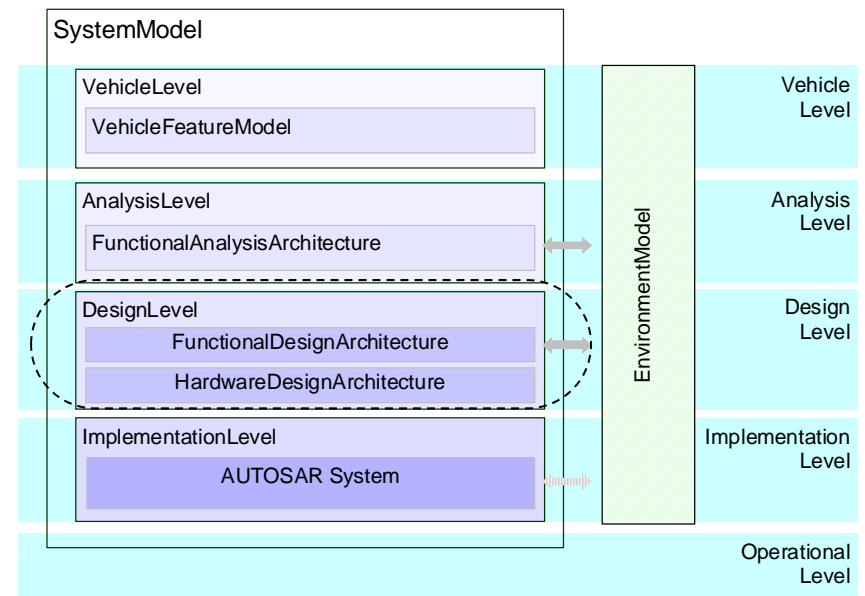


# Design Level



Design Level captures the concrete functional definition with a close correspondance with the final implementation

- Captures functional definition of application software
- Captures functional abstraction of hardware and middleware
- Captures abstract hardware architecture
- Defines Function-to-hardware allocation

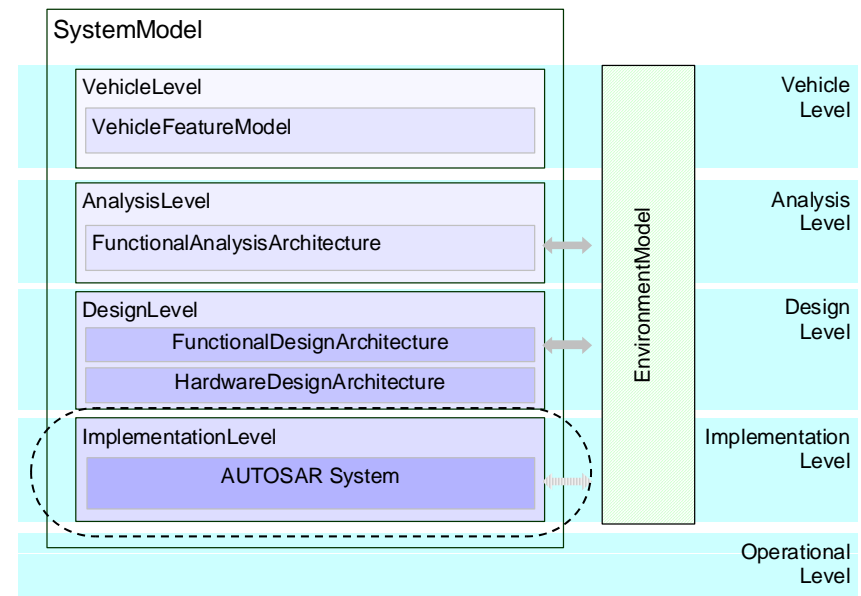


# Implementation Level



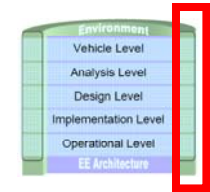
The Implementation Level represents the software-based implementation of the system

- Software components represent application functionality
- AUTOSAR Basic software represents platform
- ECU specifications and topology represent hardware
- Model is captured in AUTOSAR
- Software component template
- ECU resource template
- System Template



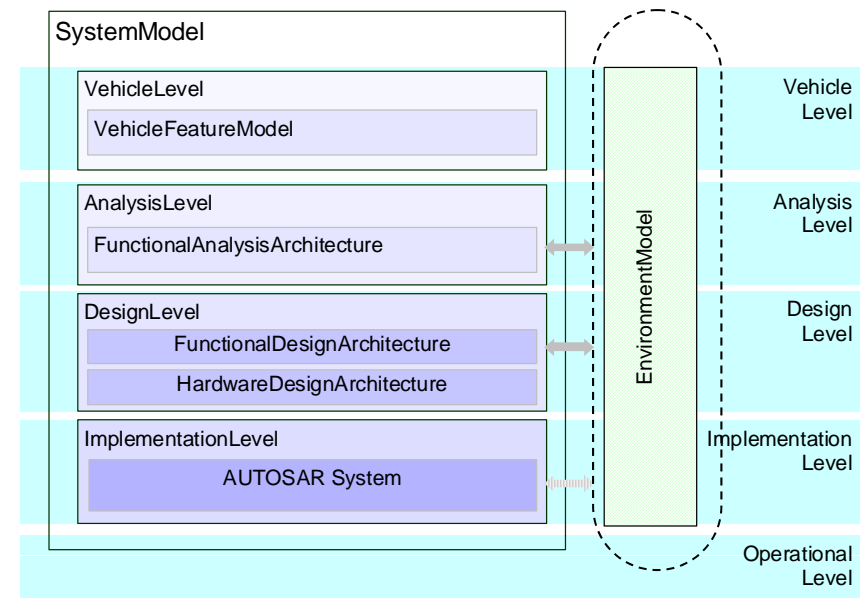


# Environment Model



The Environment model captures the plant that the EE system control and interact with

- In-vehicle, near and far environment is covered
- Same Environment Model may be used on all abstraction levels
- Different Environment models may be used depending on validation scenario



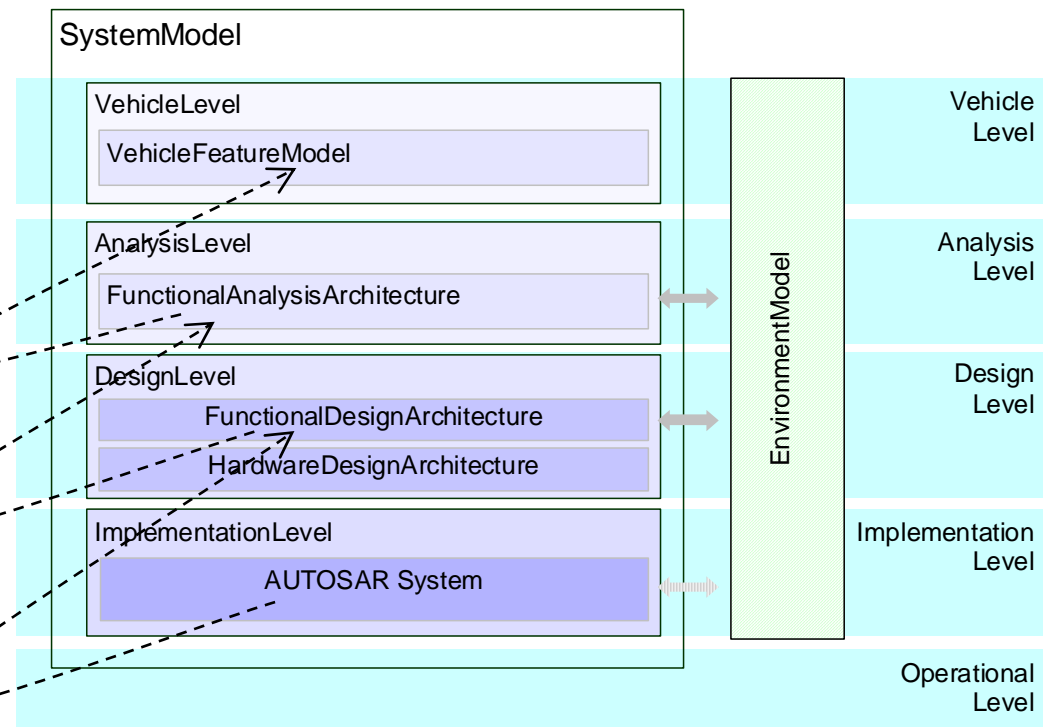
# Traceability between abstraction levels

Realization relations identify which abstract element is realized by a more concrete entity

*Functions on analysis level realizes features on vehicle level*

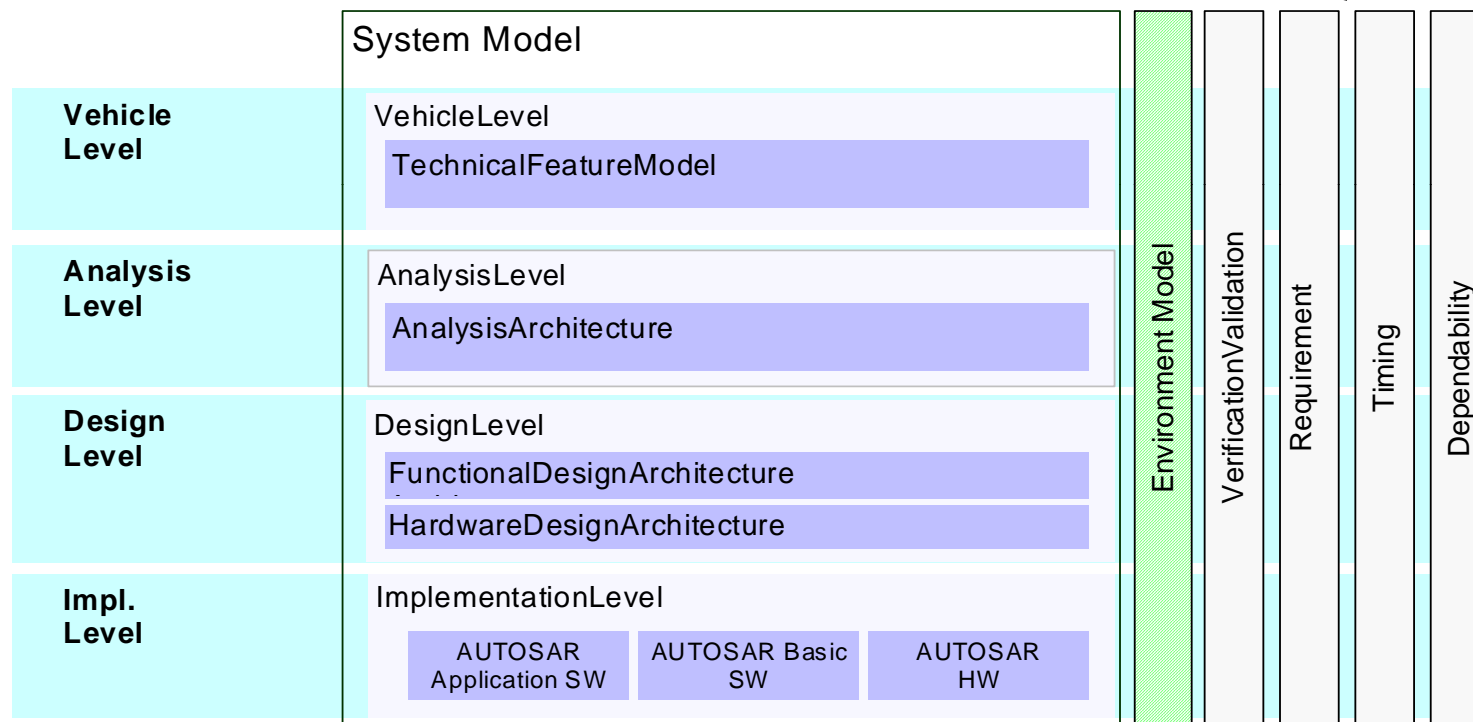
*Functions on design level realizes functions on analysis level*

*SW components or runnables on implementation level realizes functions on design level*



# Extensions

*Elements in extensions reference elements in "core"*



# EAST-ADL2 Characteristics

Extended compared to traditional ADL as it covers:

- Variability
- Requirements
- Safety
- Behavior
- Environment Modelling
- Design methodology

## EAST-ADL2

- Language Metamodel
- UML2 Profile
- Prototype Toolset

EAST-ADL has been developed in:

- EAST-EAA (ITEA 2000-2004)
- ATESSST (EC FP6 2006-2008)
- ATESSST2 (EC FP7 2008-2010)
- TIMMO (ITEA 2007-2009)

Alignment/integration:

- (SysML, AADL)
- UML/MARTE
- AUTOSAR
- ISO26262

# EAST-ADL Contributors 2000-2009

*---including*

AUDI

BMW

Carmeq /VW

CRF/FIAT

Daimler

ETAS

Mecel

Mentor Graphics

OPEL

PSA

Renault

Robert Bosch

Siemens, Continental

Valeo

Vector

Volvo Car Corporation

Volvo Technology

ZF

CEA-LIST

INRIA

LORIA

Paderborn Univerisity-C-LAB

Technical University of Darmstadt

Technische Universität Berlin

The Royal Institute of Technology

The University of Hull

...

# Relation to other modeling languages and approaches?

## Why Not UML?

- EAST-ADL2 is domain-specific but its UML2 profile gives access to UML2 tools.

## Why not SysML?

- EAST-ADL takes up applicable SysML concepts but provides additional domain-specific support

## Why not AUTOSAR?

- EAST-ADL complements AUTOSAR with respect to feature content, functional structure, safety properties, etc.

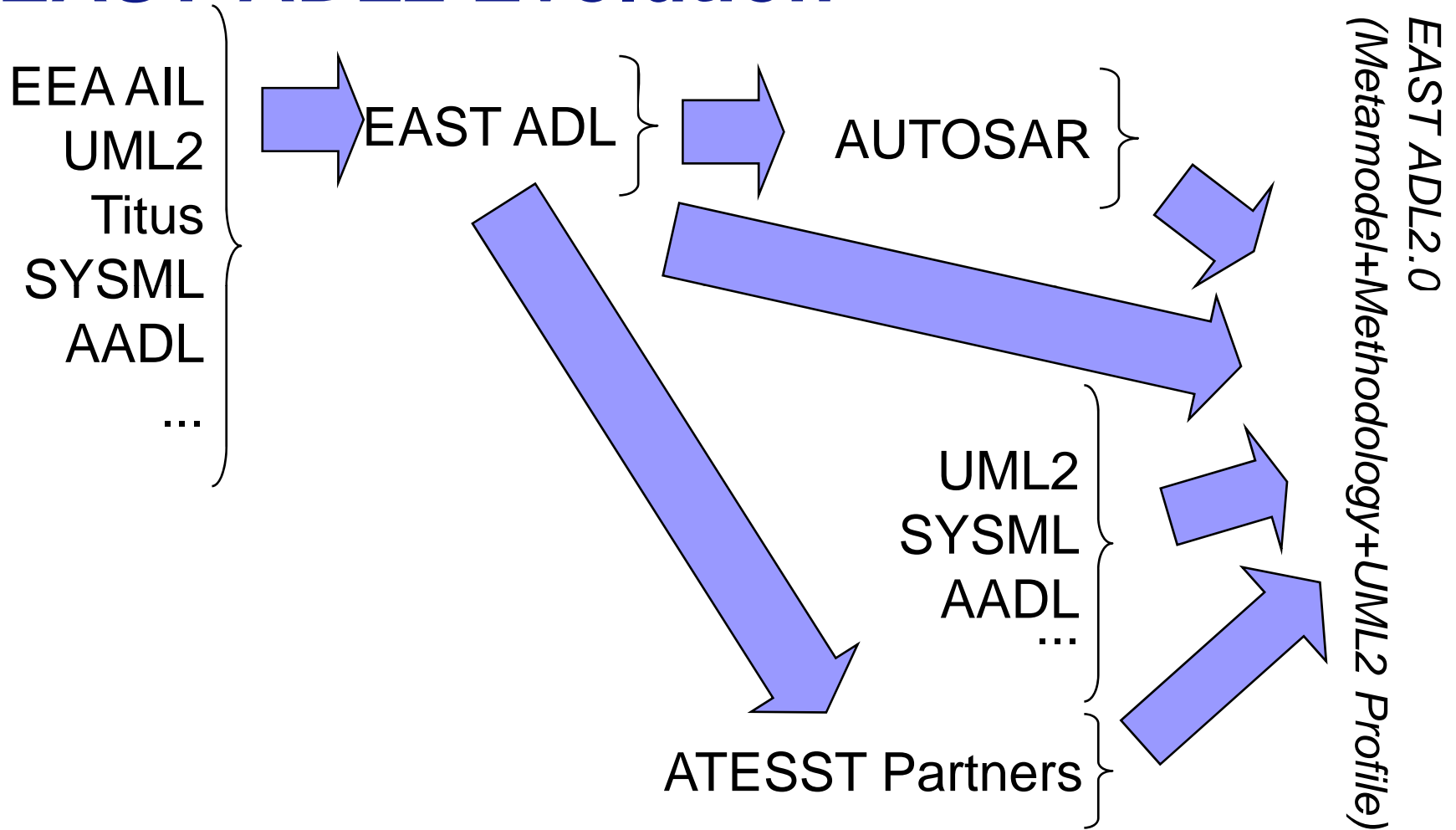
## Why not AADL

- AADL represent the software implementation of a system while EAST-ADL2 starts on a more abstract level.

## Why not proprietary tools (Simulink, Statemate, Dymola, ASCET, ...)?

- EAST-ADL2 provides an information structure for the engineering data and integrates external tools

# EAST-ADL2 Evolution



# EAST-ADL2 Complements AUTOSAR

EAST-ADL2 is an information structure including aspects beyond the Software Architecture

Requirements, traceability, feature and function content, variability, etc.

Provides means to define what the software does

An AUTOSAR specification defines the software architecture and information required for SW integration - but is neutral to its functionality

Provides means to model strategic properties

Key vehicle aspects is captured independently of the software architecture

Supports modelling of error behavior and the representation of safety-related information and requirements



# EAST-ADL2 Tooling

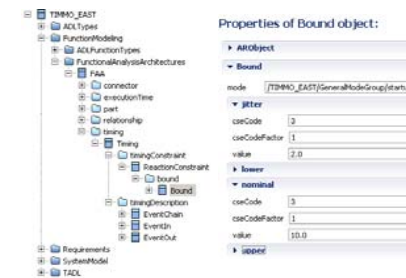
## UML-based Tooling

- Based on CEA Papyrus
- Integrated Eclipse application with 5 ATESST plugins



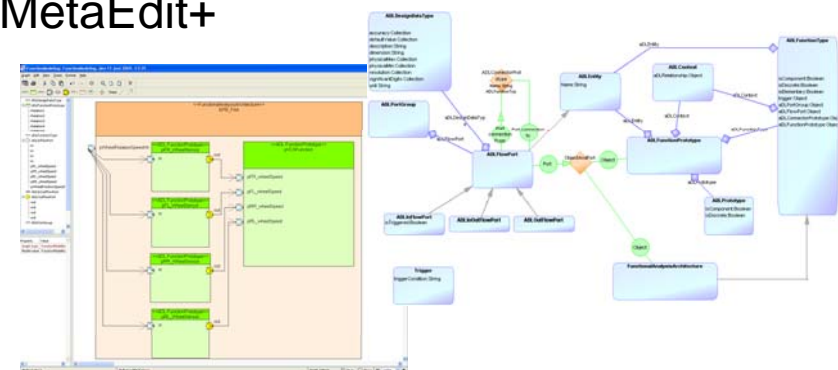
## AUTOSAR-based Tooling

- MentorGraphics VSA



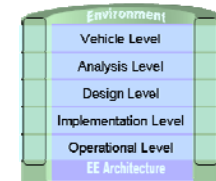
## DSL Tooling

- MetaEdit+



# Conclusion

## EAST-ADL2



EAST-ADL2 provides an information structure for design of automotive embedded systems

- Architecture Description Language and Architectural Framework

Use of abstraction levels is a fundamental concept

- entities on lower levels *realize* entities on higher levels

EAST-ADL2 is a fully aligned complement to AUTOSAR

- AUTOSAR is the SW architecture definition enabling SW component integration on ECU
- EAST-ADL2 supports the successful integration of AUTOSAR components
- EAST-ADL2 Supports additional engineering steps including ***feature definition, requirements engineering, V&V, safety analysis, functional modeling/integration, product line engineering***