



The AUTOSAR Way of Model-Based Engineering of Automotive Systems

Heiko Dörr
International Conference on Graph Transformation
Leicester, 12.09.2008

Purpose of the Presentation.

Long history of graph transformation

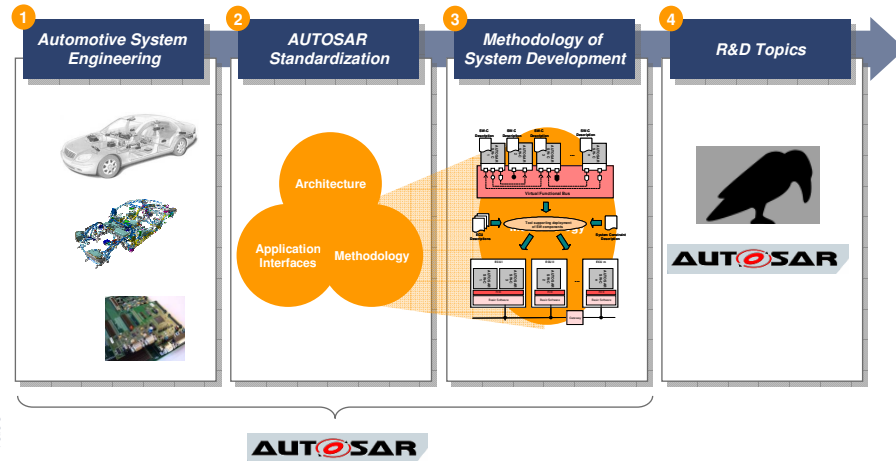
- Sound body of theory
- Several areas of application
- Tool-support available to handle larger models

Future model-based engineering of automotive systems will use AUTOSAR

Explore potentials of the application of graph transformation in an AUTOSAR development

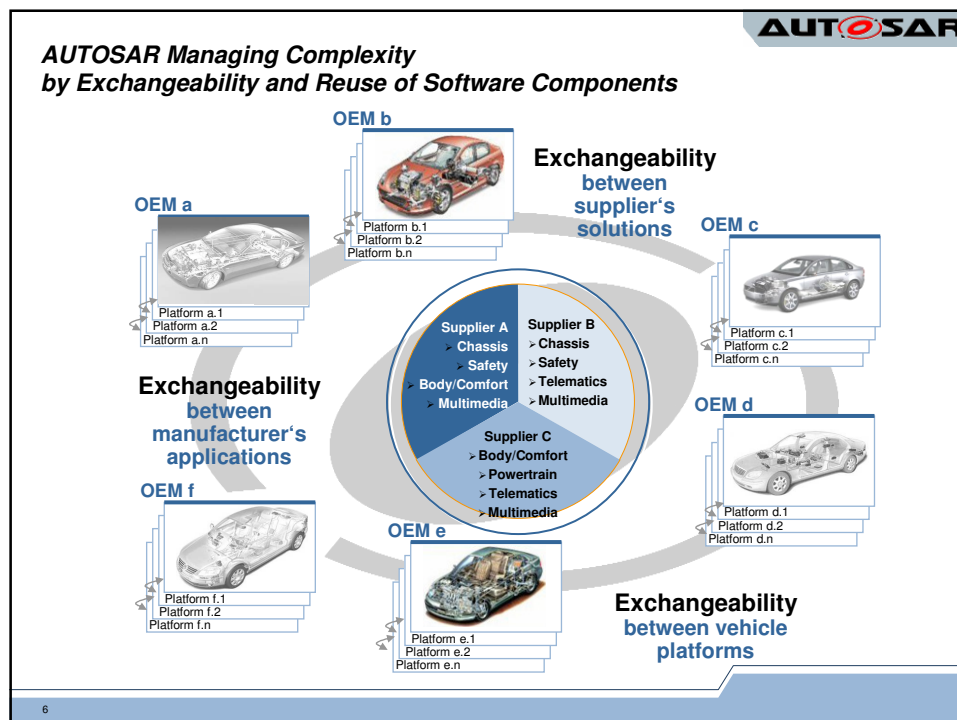
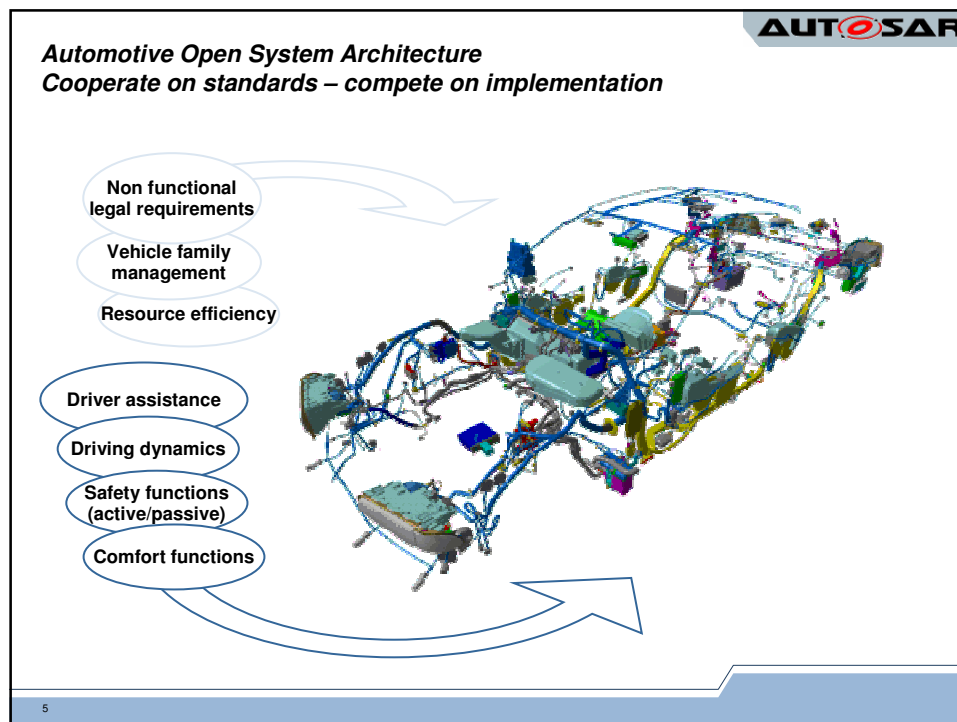
Identification of research and development topics

Content.

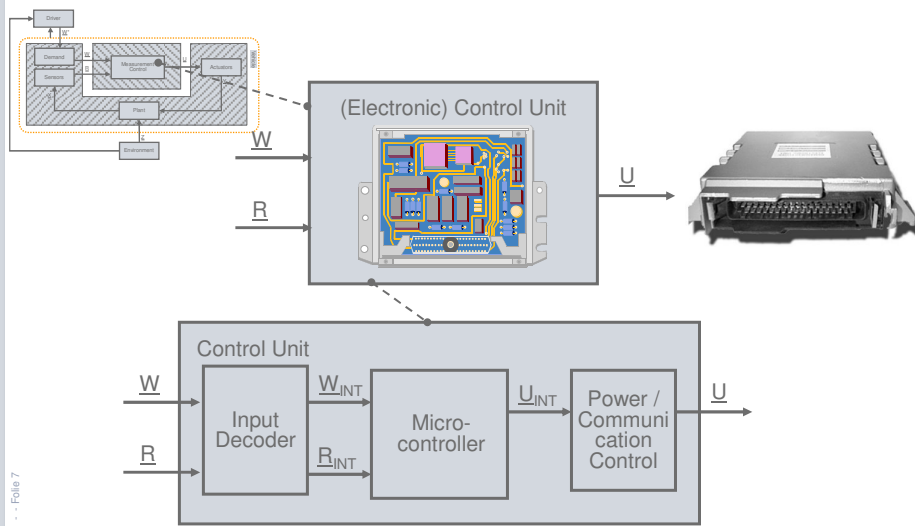


- Folie 3

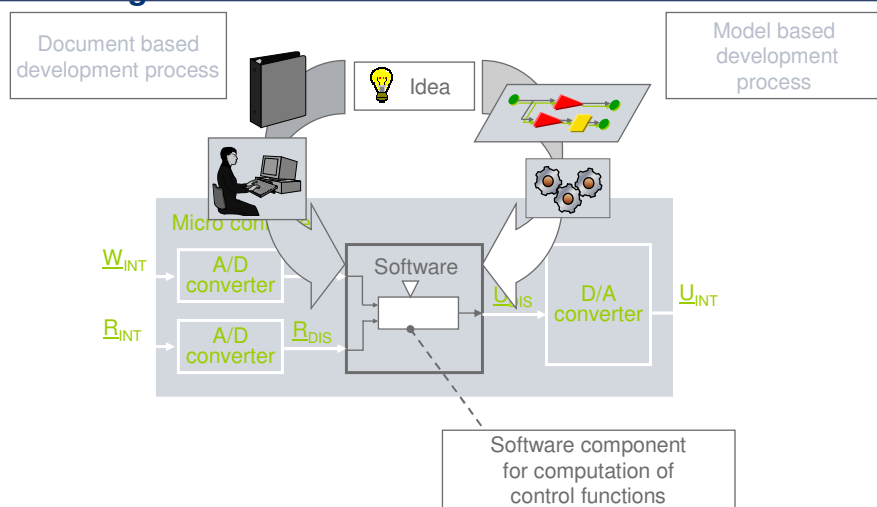
Automotive Systems and SW Engineering

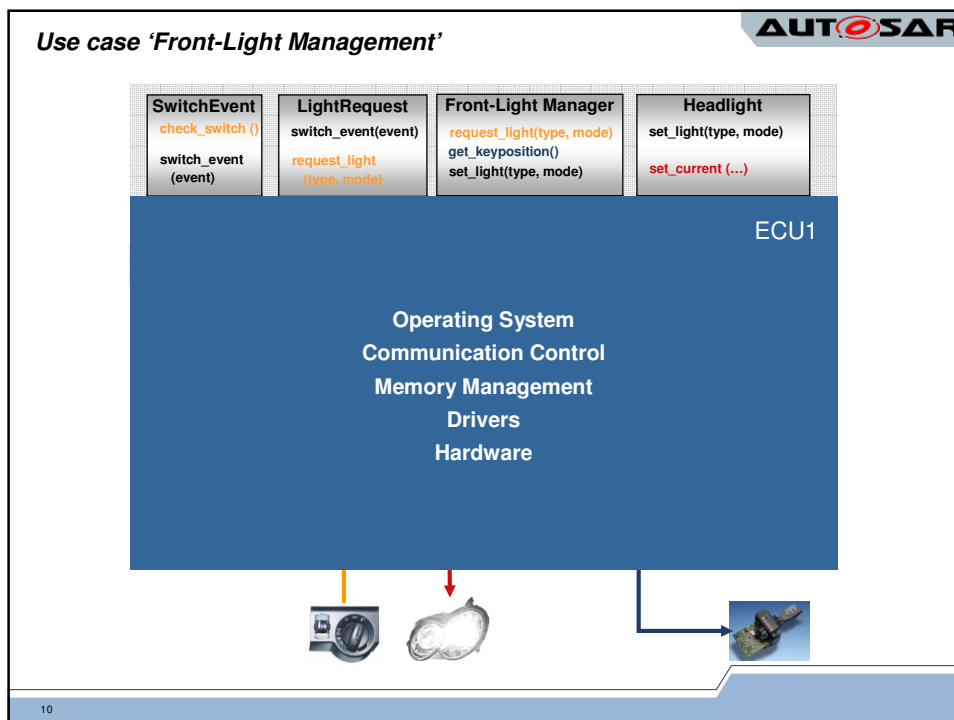
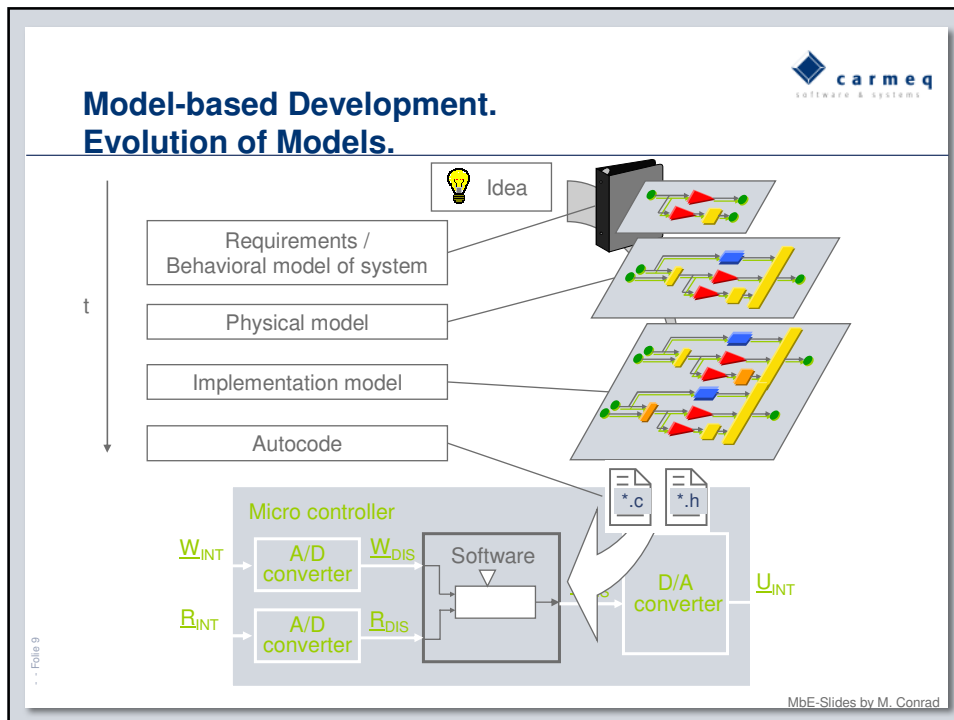


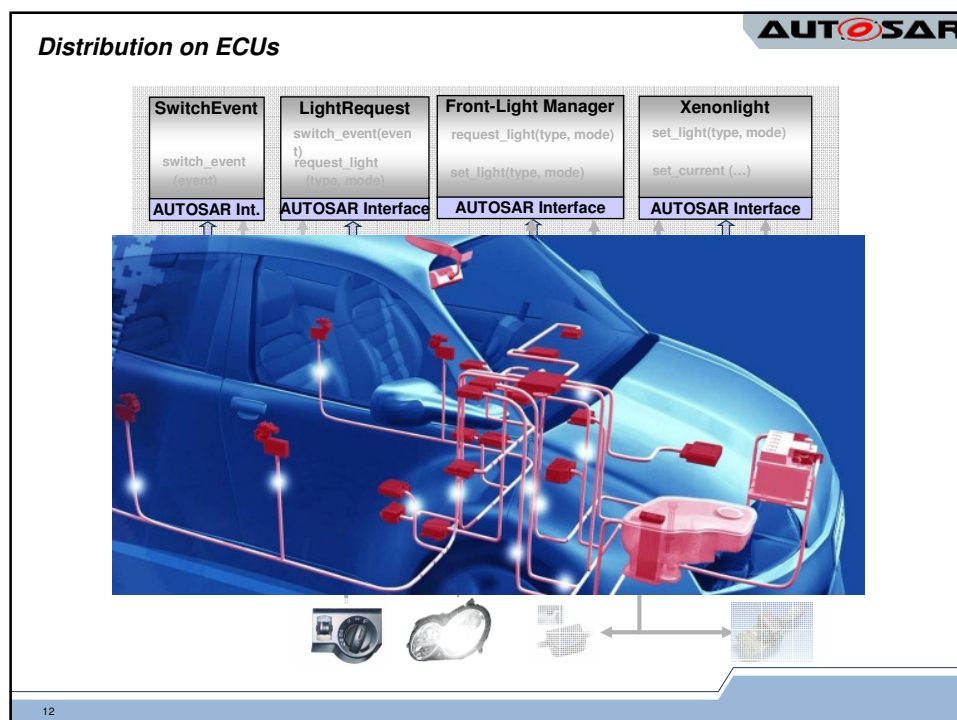
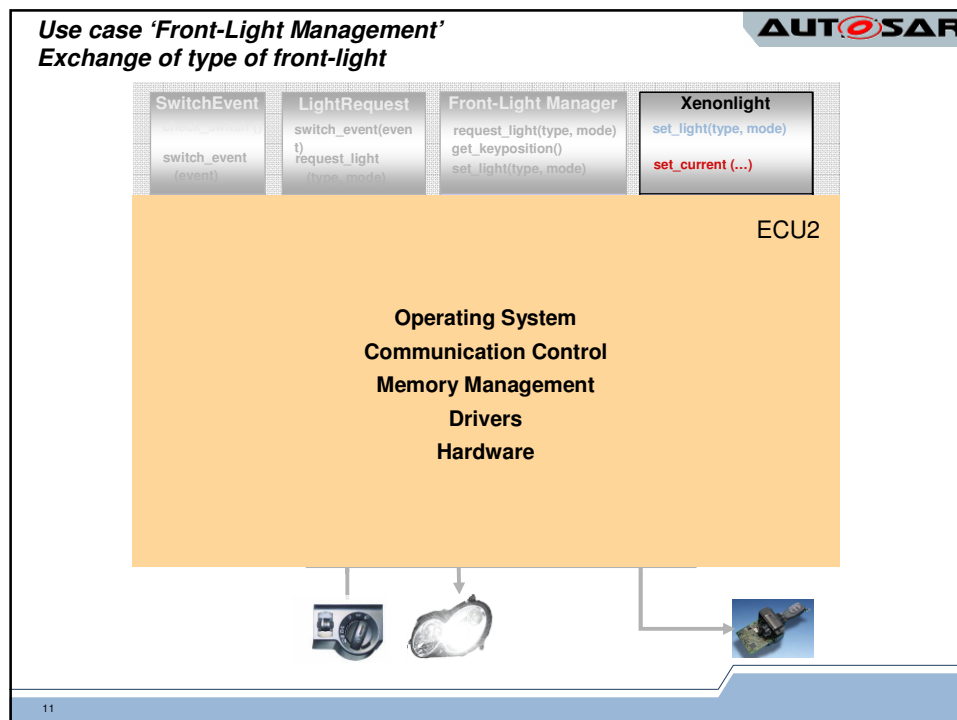
Model-based Development. General Task.

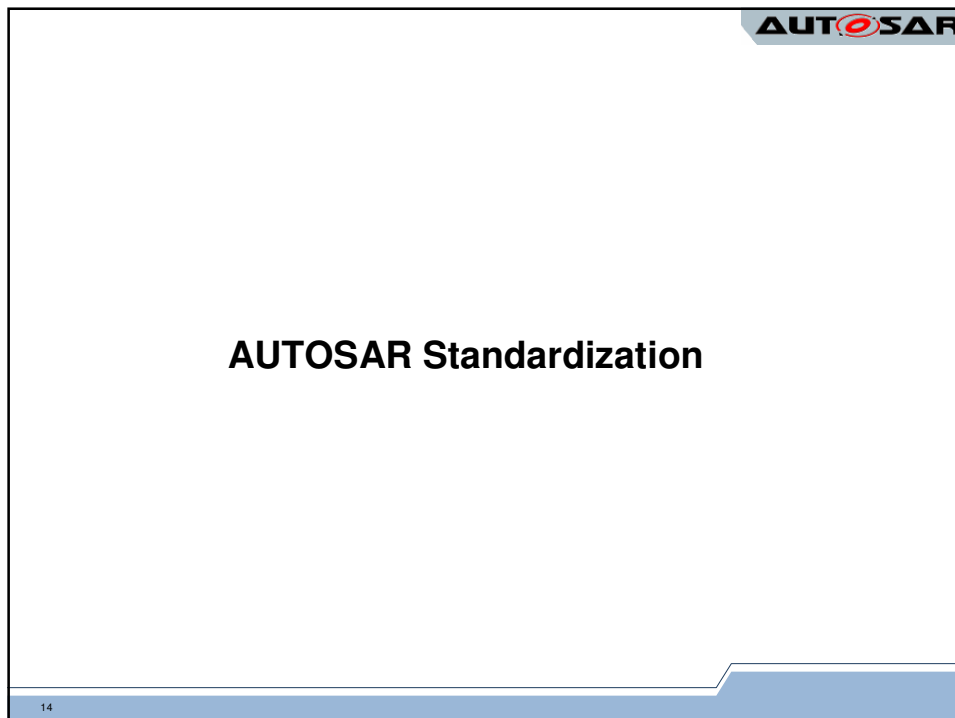
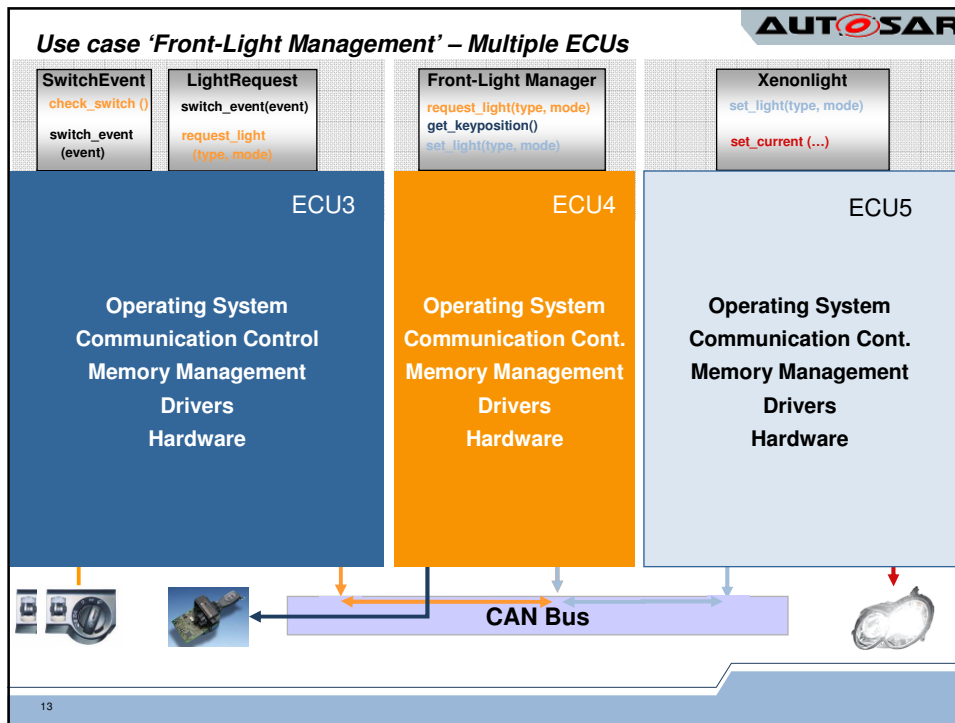


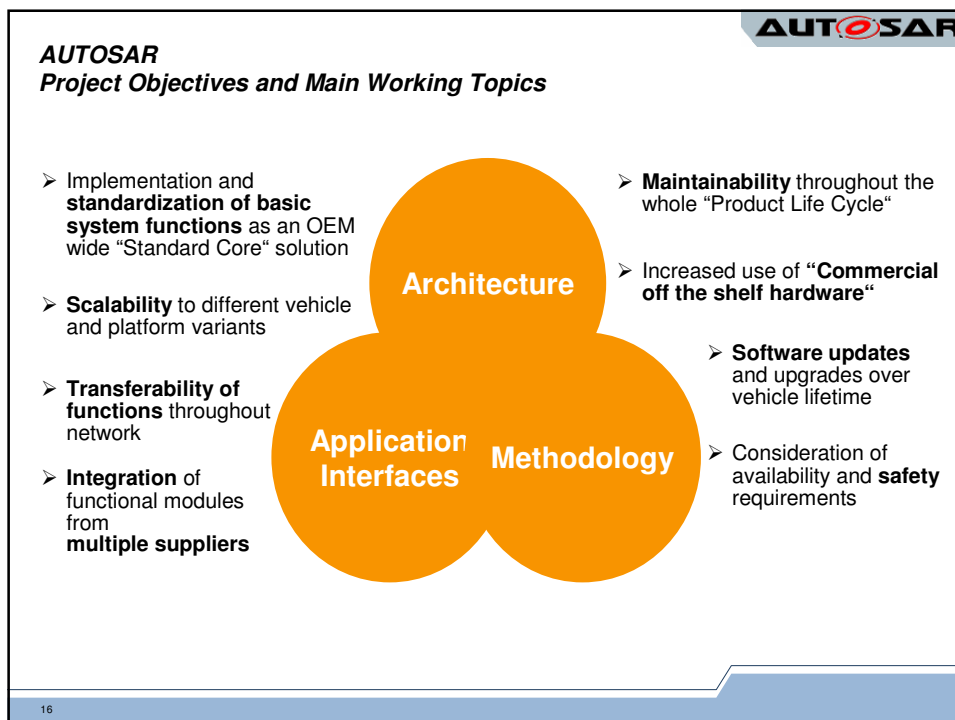
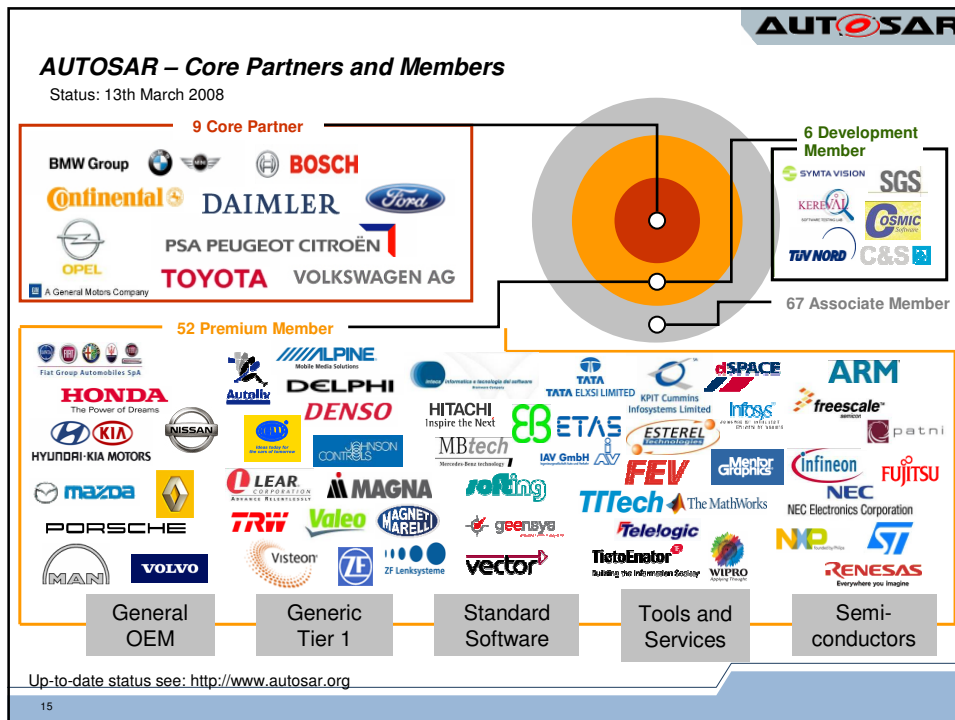
Model-based Development. Paradigm Shift.



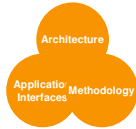




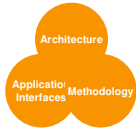




AUTOSAR Main Working Topics



- **Architecture:**
Software architecture including a complete basic or environmental software stack for ECUs – the so called AUTOSAR Basic Software – as an integration platform for hardware independent software applications.



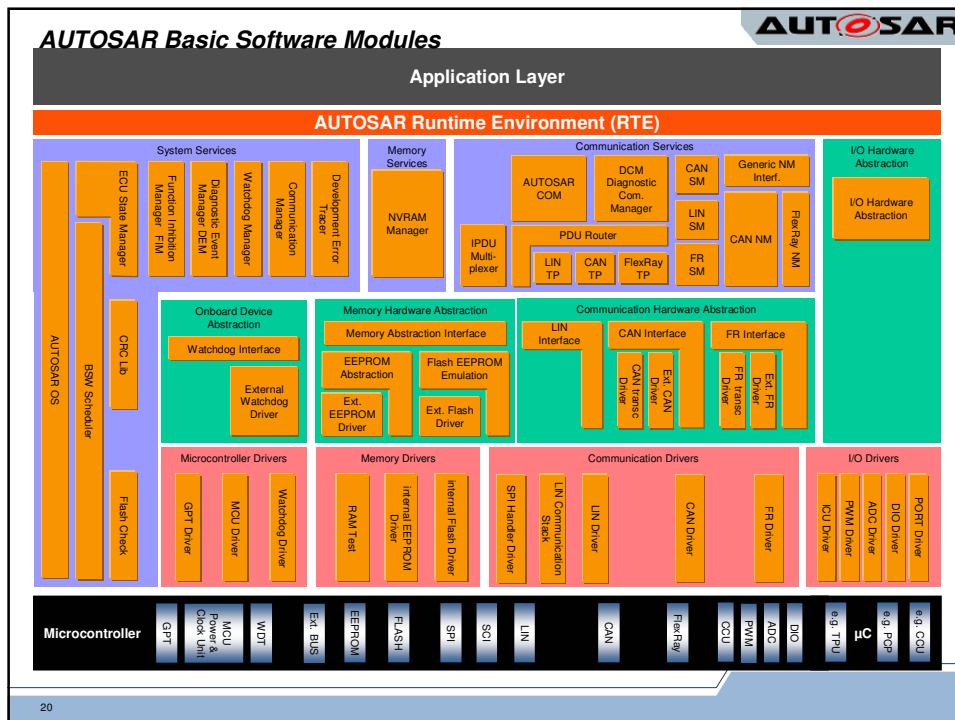
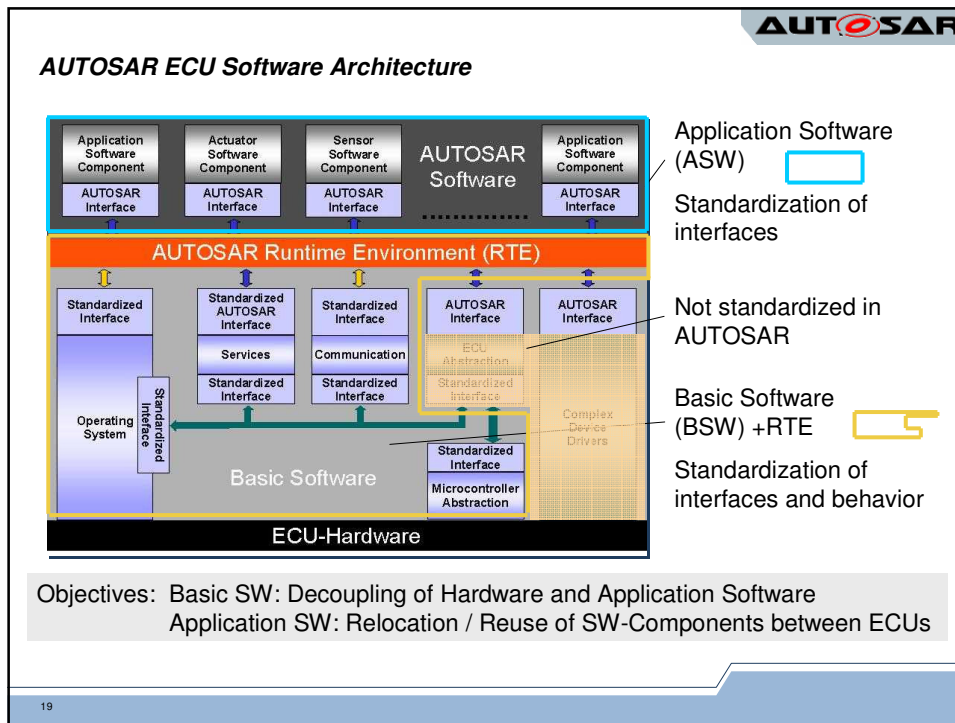
- **Methodology:**
Exchange formats or description templates to enable a seamless configuration process of the basic software stack and the integration of application software in ECUs and it includes even the methodology how to use this framework.



- **Application Interfaces:**
Specification of interfaces of typical automotive applications from all domains in terms of syntax and semantics, which should serve as a standard for application software.

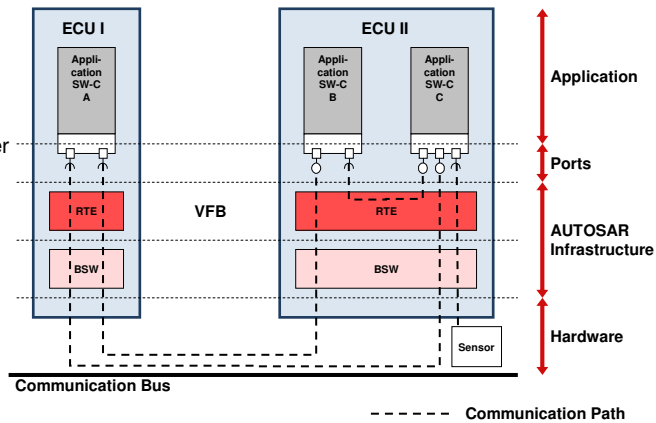
Main Concepts: Architecture

- Basic Software modules
- Run time environment and communication



Intra- and Inter-ECU Communication

- Ports implement the interface according to the communication paradigm (here client-server based).
- Ports are the interaction points of a component.
- The communication is channeled via the RTE.
- The communication layer in the basic software is encapsulated and not visible at the application layer.



21

AUTOSAR Architecture – Conclusion

- 1** AUTOSAR harmonizes already existing basic software solutions and closes gaps for a seamless basic software architecture.
- 2** AUTOSAR aims at finding the best solution for each requirement and not finding the highest common multiple.
- 3** The decomposition of the AUTOSAR layered architecture into some 40 modules has proven to be functional and complete.

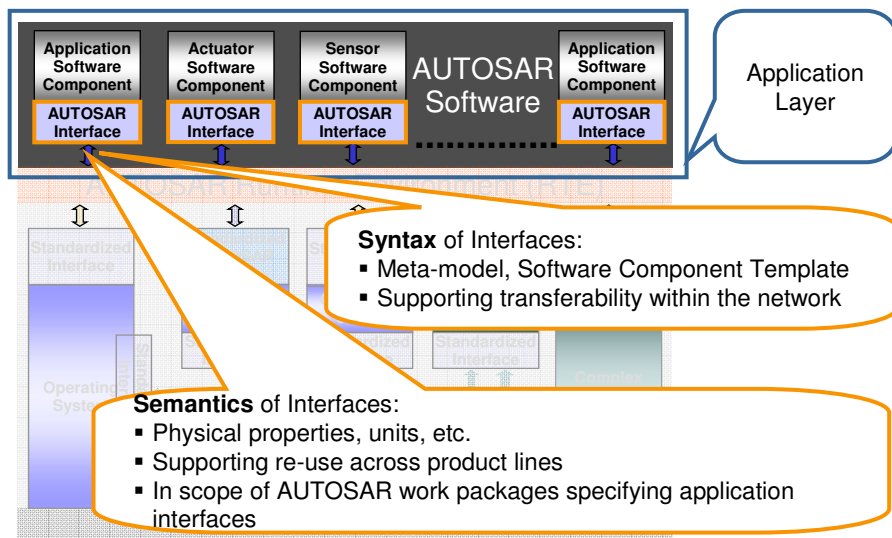
22

Main Concepts: Application Interfaces

- Standardization approach
- Current stage of standardization

23

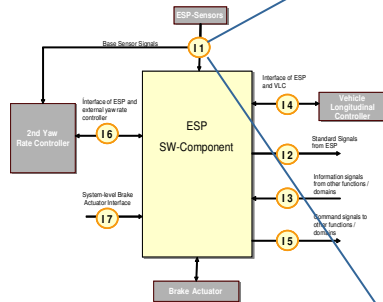
AUTOSAR Application Interfaces



24

To ease the re-use of software components across several OEMs, AUTOSAR proceeds on the standardization of the application interfaces agreed among the partners.

Example

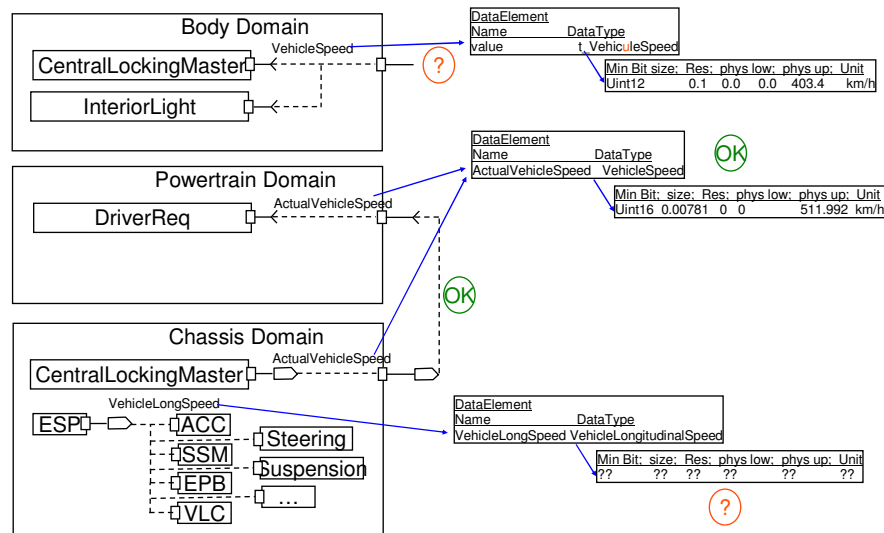


Standardized application interfaces on system level
(ESP-system, chassis domain)

| | |
|-----------------|--|
| Data Type Name | LongAccBase |
| ... | |
| Data Type Name | YawRateBase |
| Description | Yaw rate measured along vehicle z- axis (i.e. compensated for orientation). Coordinate system according to ISO 8855 |
| Data Type | S16 |
| Integer Range | -32768...+32767 |
| Physical Range | -2,8595...+2,8594 |
| Physical Offset | 0 |
| Unit | rad/sec |
| ... | ... |
| Remarks | This data element can also be used to instantiate a redundant sensor interface. Range might have to be extended for future applications (passive safety). |
| ... | |
| Data Type Name | RollRateBase |

25

Major task: Conflict Resolution – Example Vehicle Speed



26

AUTOSAR Application Interfaces – Conclusion

1

For several domains a subset of application interfaces has been standardized to agreed levels.

2

It is a challenge to align standardization with the pace of application development.

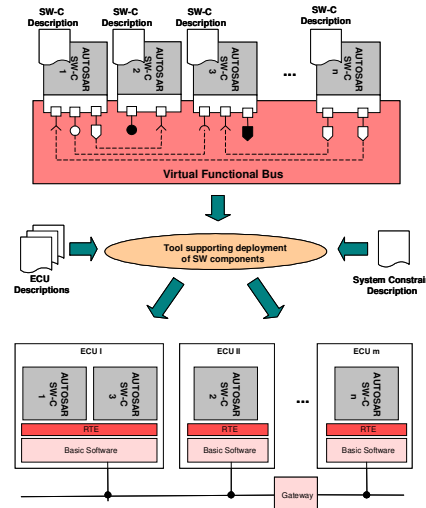
27

Main Concepts: Methodology

- Overall methodology
- Structure of configuration information
- System Design – Implementation Process

28

Following the AUTOSAR Methodology, the E/E architecture is derived from the formal description of software and hardware components.

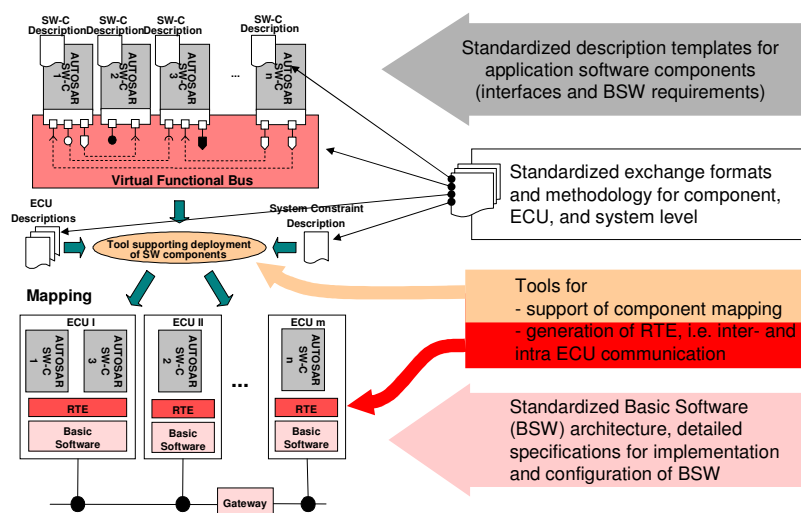


- Functional software is described formally in terms of "Software Components" (SW-C).
- Using "Software Component Descriptions" as input, the „Virtual Functional Bus“ validates the interaction of all components and interfaces before software implementation.
- Mapping of "Software Components" to ECUs and configuration of basic software.
- The AUTOSAR Methodology supports the generation of an E/E architecture.

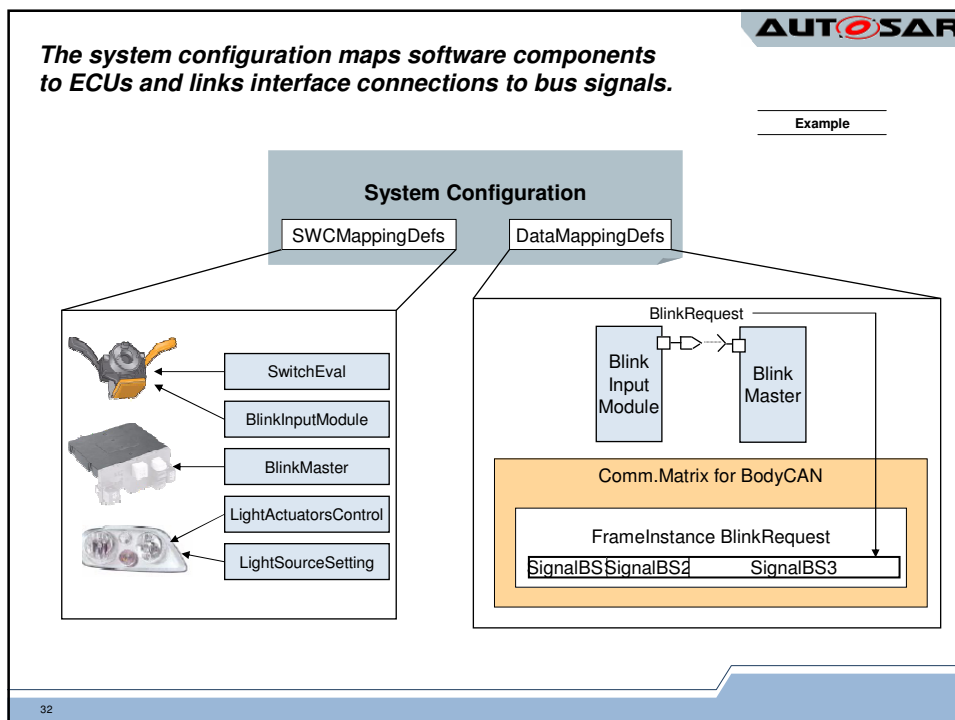
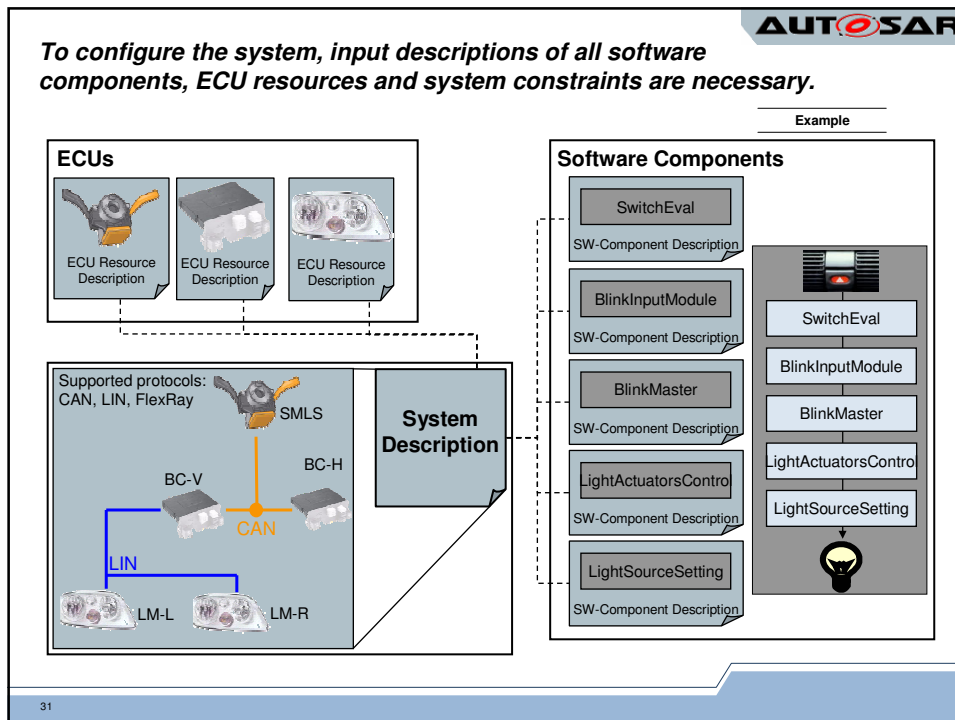
29

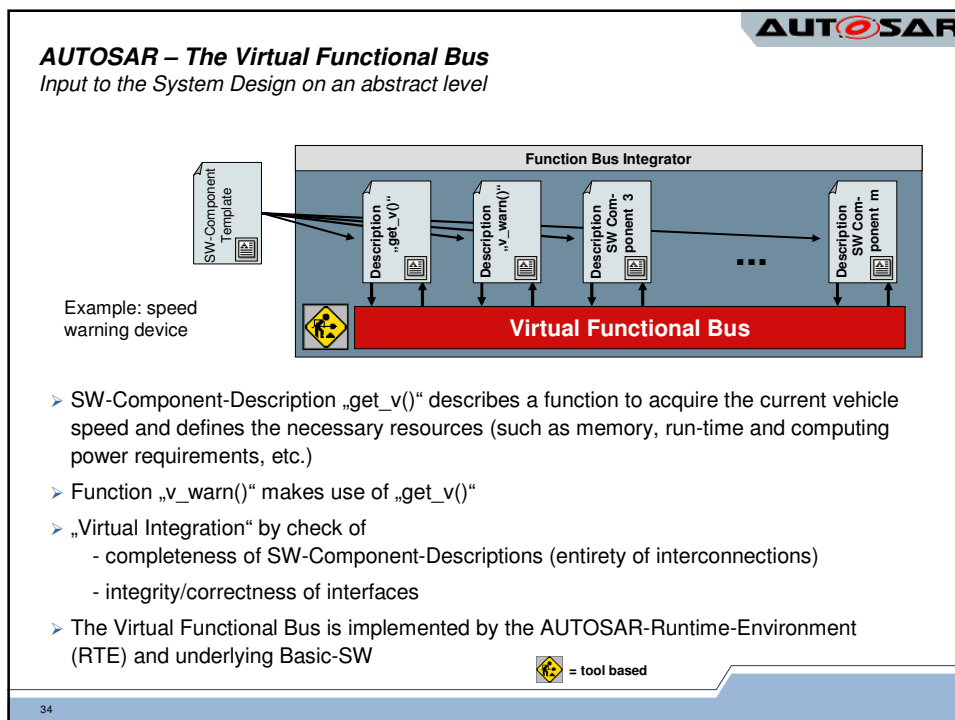
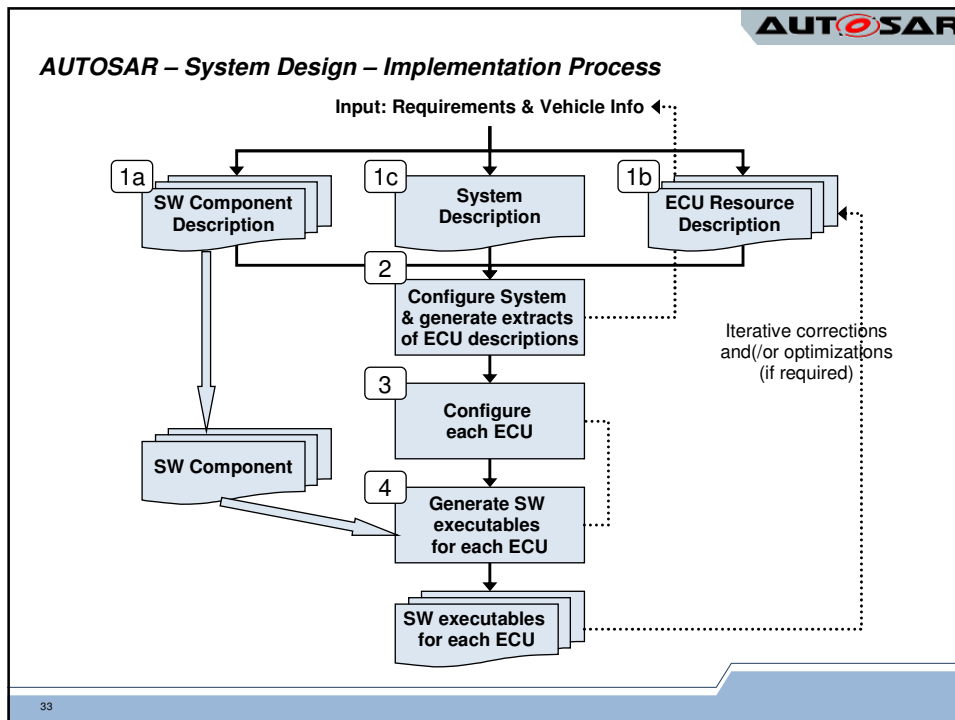
AUTOSAR Methodology
Derive E/E architecture from formal descriptions of soft- and hardware components

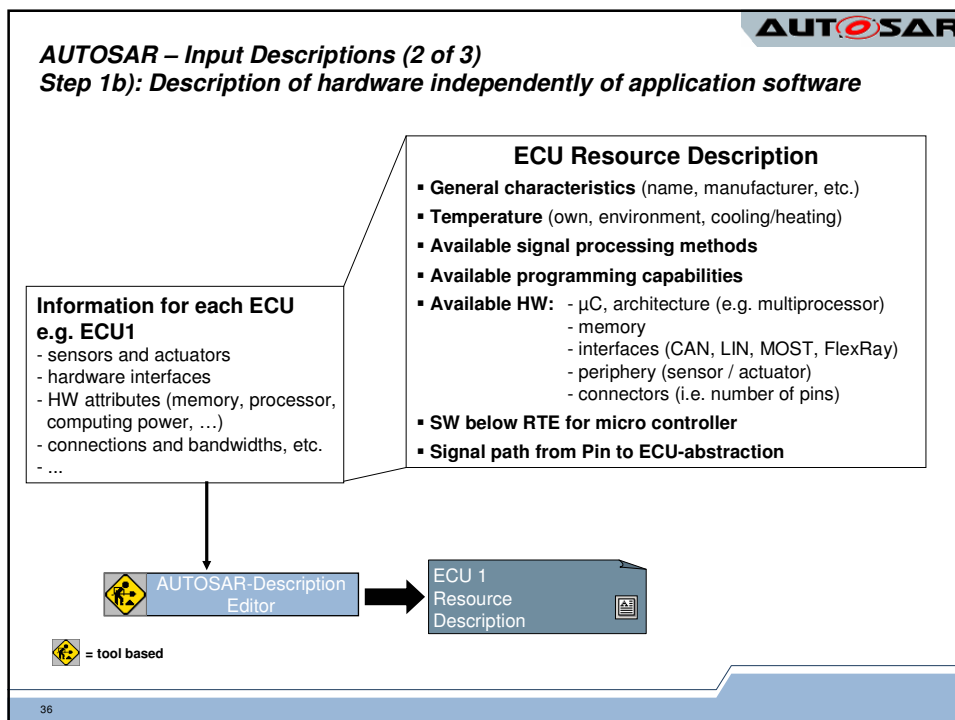
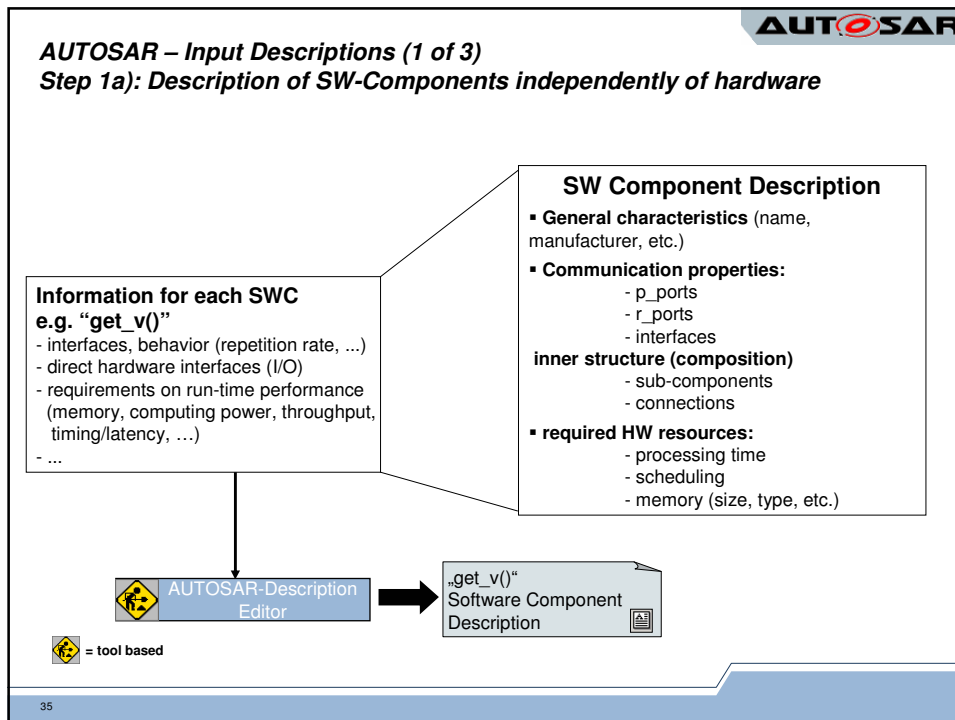
VFB view



30

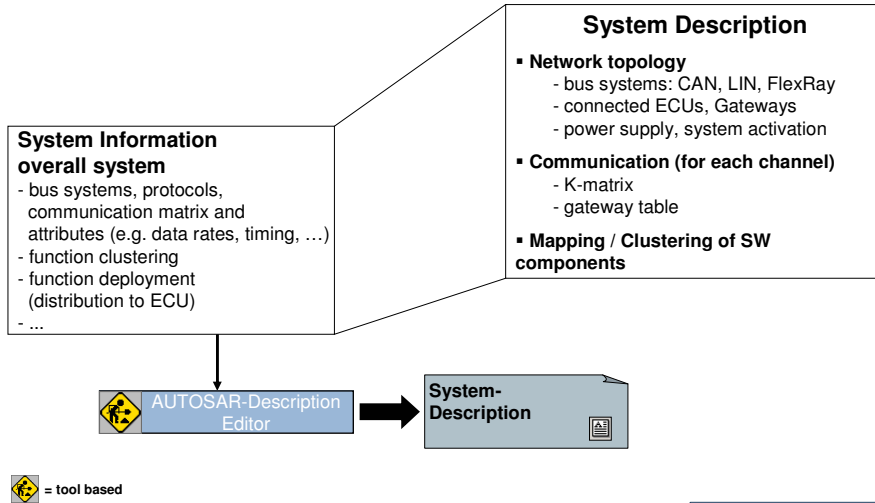






AUTOSAR – Input Descriptions (3 of 3)

Step 1c): Description of system

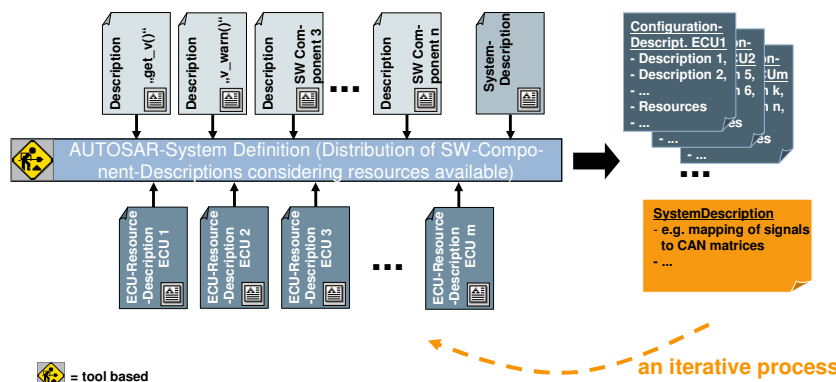


37

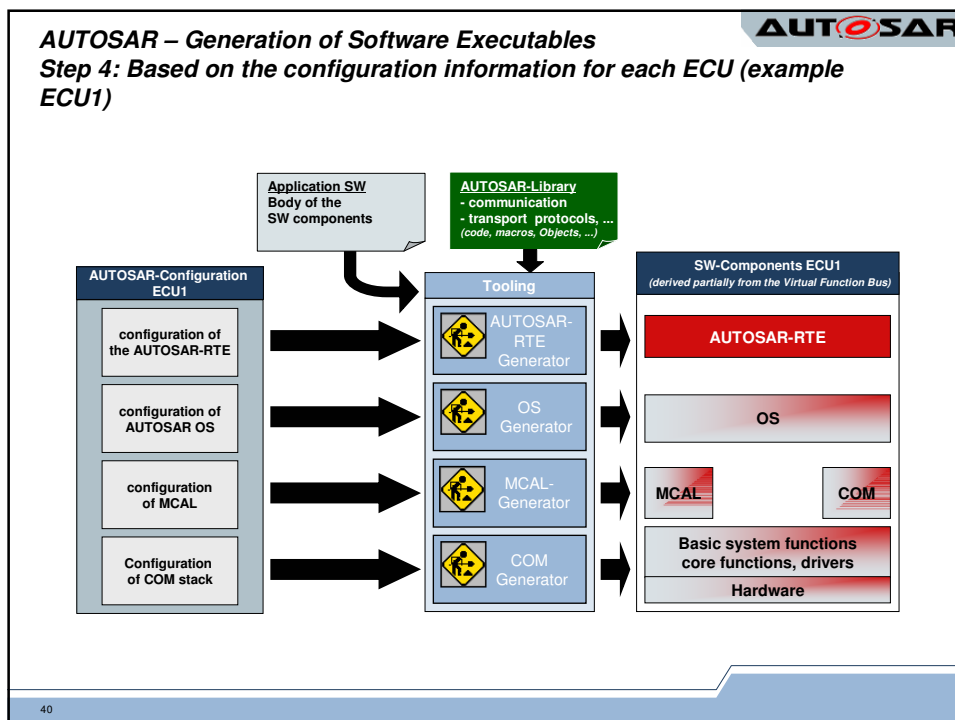
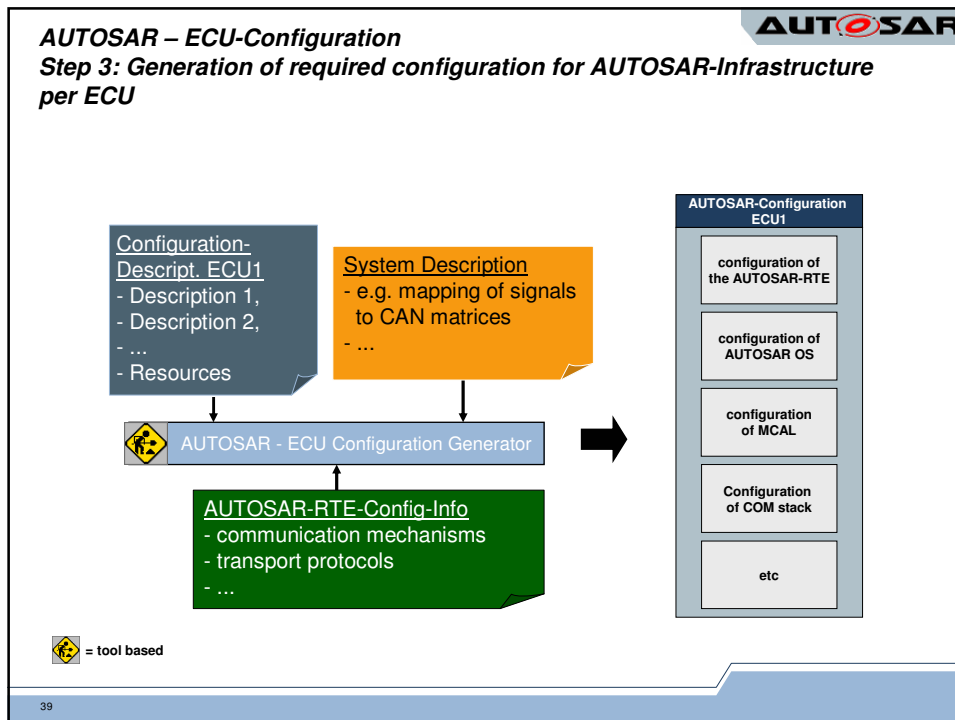
AUTOSAR – System Configuration

Step 2: Distribution of SW-Component-Descriptions to ECU

- Configuration on the basis of descriptions (not on the basis of implementations!) of SW-Components, ECU-Resources and System-Description
- Consideration of ECU-Resources available and constraints given in the System-Description



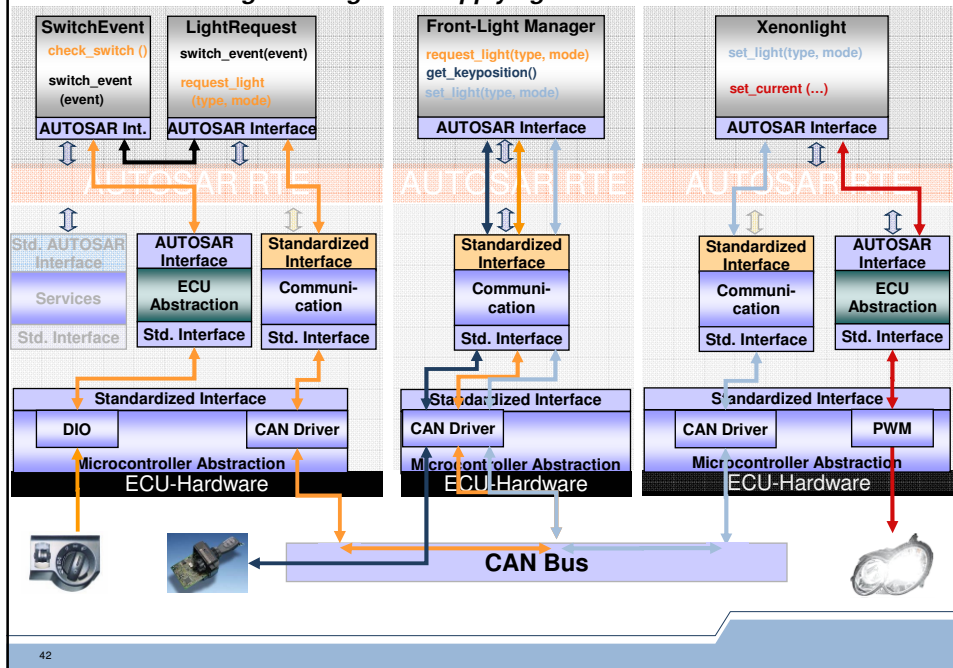
38



AUTOSAR Methodology – Conclusion

- 1 The E/E system architecture can be described by means of AUTOSAR.
- 2 The meta model approach and the tool support for specifying the AUTOSAR information model allow working at the right level of abstraction.
- 3 A methodology to integrate AUTOSAR software modules has been designed.
- 4 AUTOSAR pushes the paradigm shift from an ECU based approach to a function based approach in automotive software development.

Use case 'Front-Light Management' applying AUTOSAR

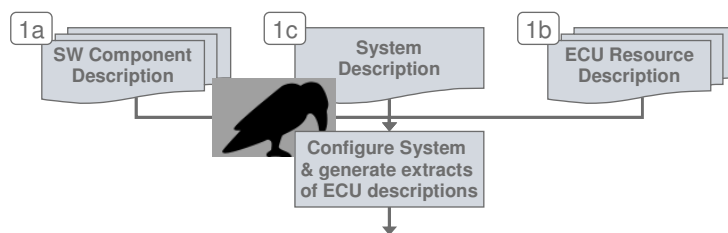


Topics for Research and Development

System configuration
Semi-automatic mapping of communication

- Folie 43

System Configuration.



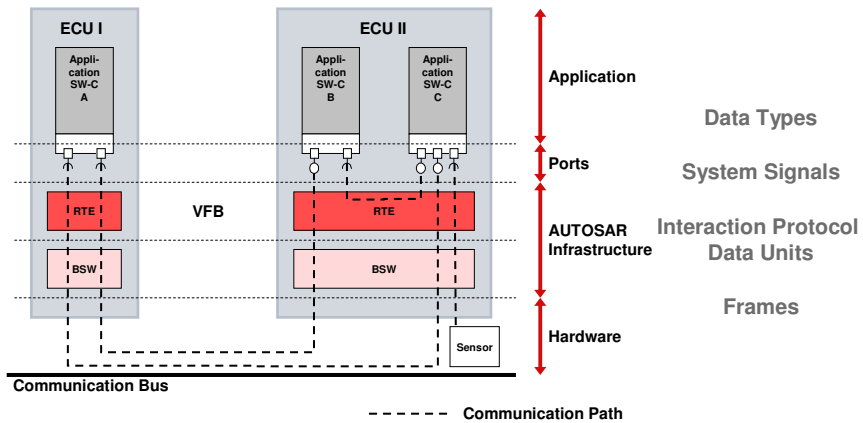
Step 2 "System Configuration" has high complexity

Contains mappings of
Data → Signals → Network Communication
Implementations ← SW-compositions → ECU
Logical HW resources → ECU
Under the existence of mappings constraints

System configuration is data structure covering the whole system description

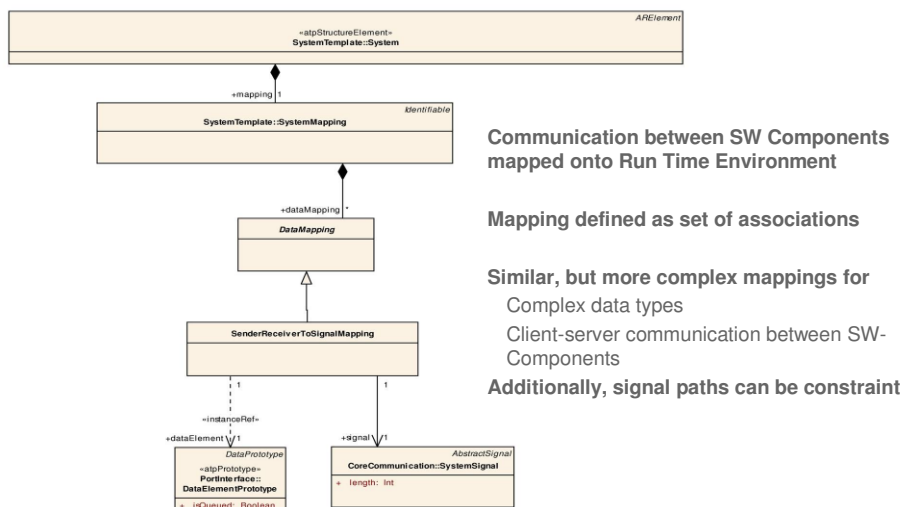
- Folie 44

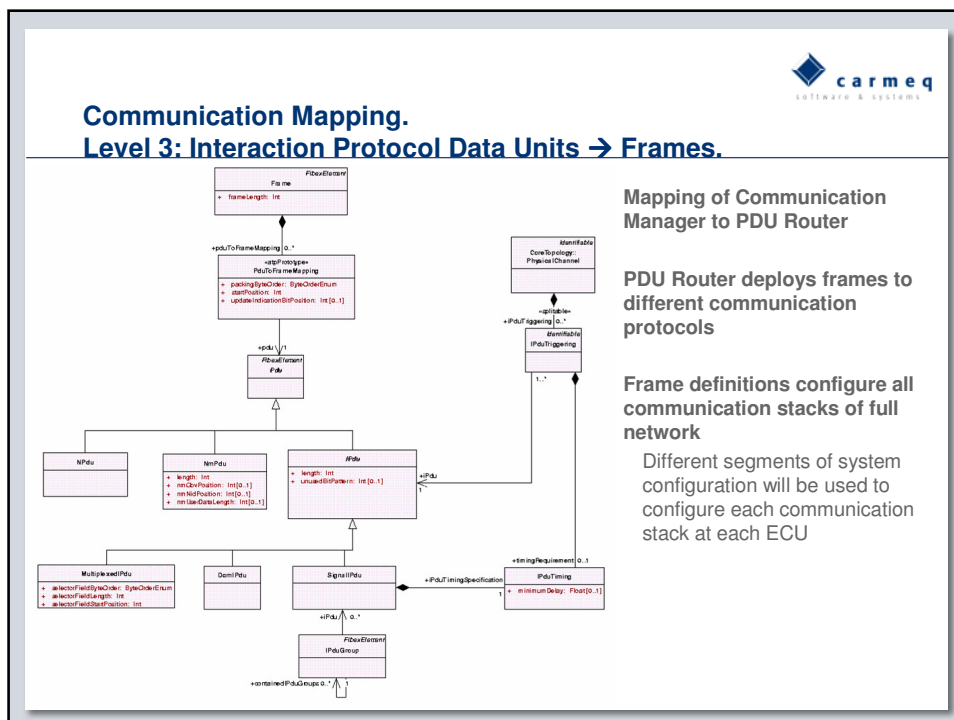
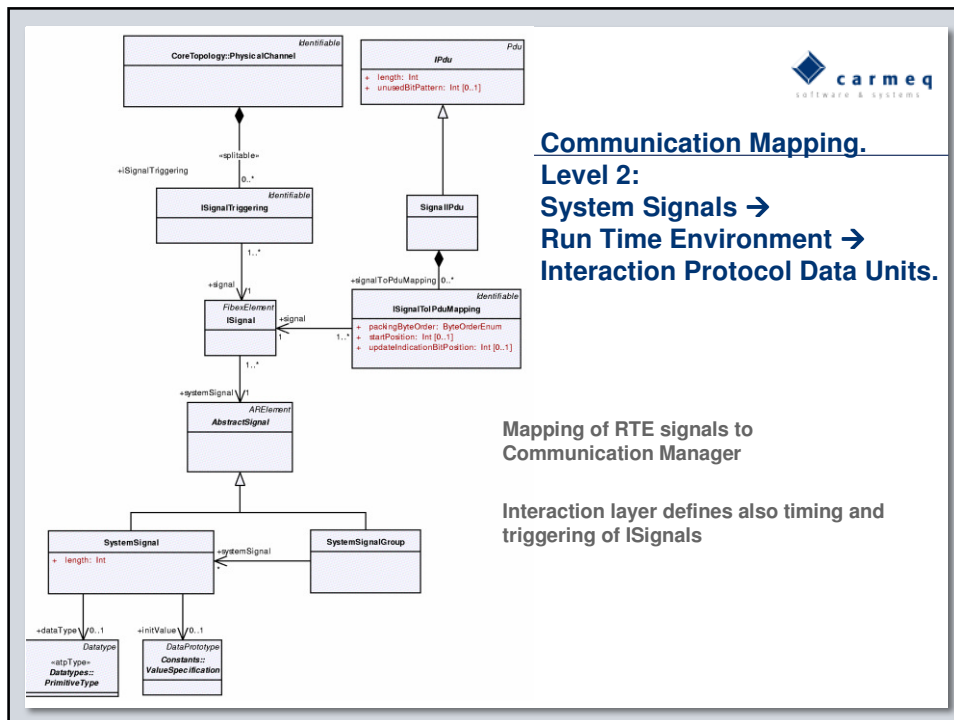
System Configuration. Communication Mapping.



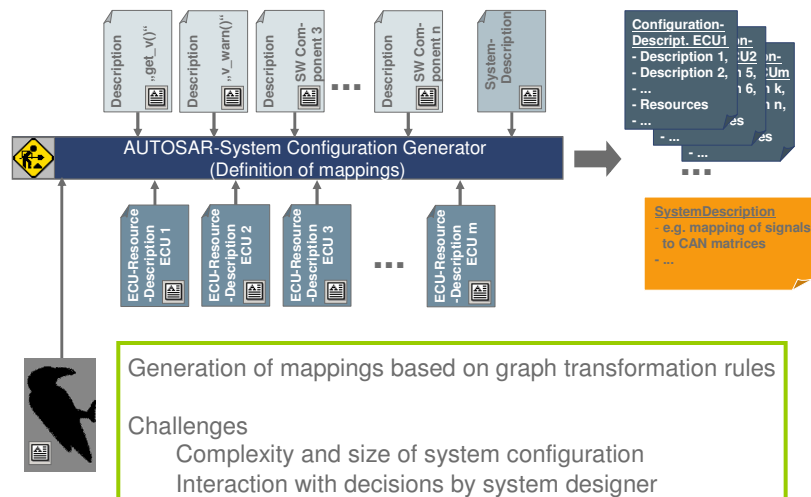
- - Folie 45

Communication Mapping. Level 1: Primitive Data Types → System Signals.





Communication Mapping Tooling.



- - Folie 49

Conclusion

AUTOSAR

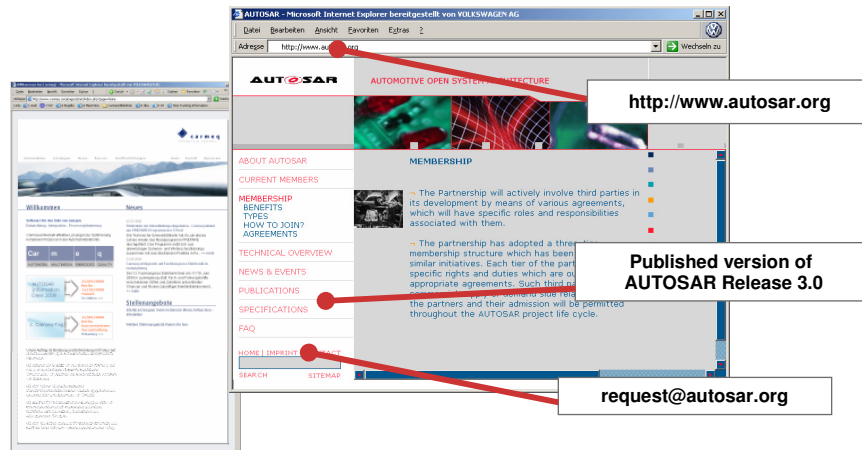
- Leverages model-based engineering of automotive software to whole systems
- Standardization itself is highly formalized and so supports formal system development
- Shifts implementation efforts to configuration

Graph Transformation

- Is the technology for semi-automatic configuration
- Can reduce the configuration complexity
- Needs to build domain knowledge

- - Folie 50

Further Information



- - Folie 51

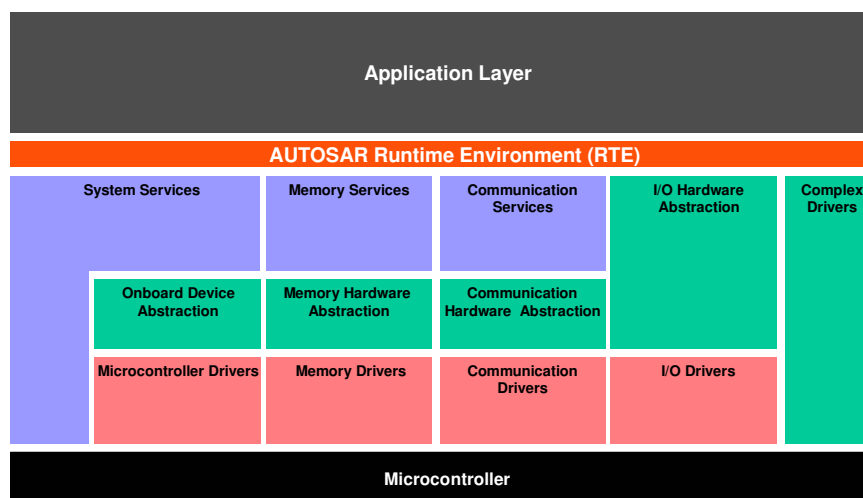
Backup

- - Folie 52

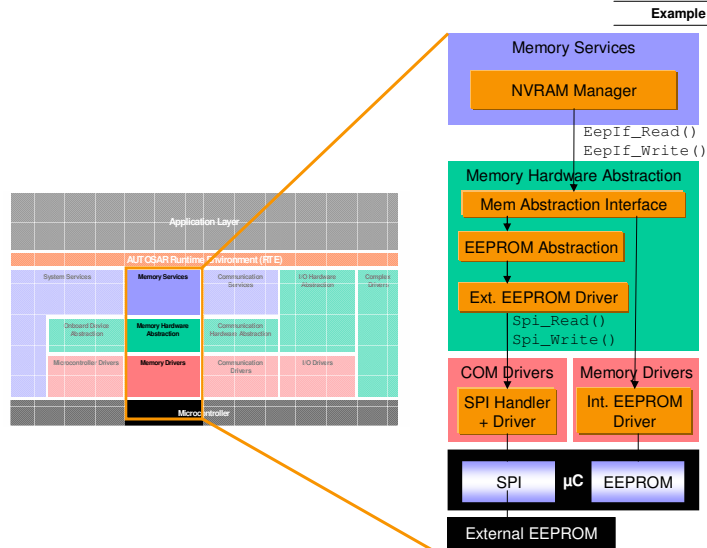
Challenges in Automotive E/E Development

- Extend product offering and increase product differentiation
 - Stable or decreasing development costs
- Strengthen brand image in the market
 - Propose specific features and functions across the product range
- Ensure long term competitiveness, as well as presence in emerging markets, through cost reduction
- Increase quality and reduce “non quality” costs
- Increasing share of electronics in vehicle value
 - Electronics share (in value): 2004: 20% → 2015: 40%
 - (McKinsey, Automotive Electronics - Managing innovations on the road)
 - Software share (in value): 2000: 4,5% → 2010: 13%
 - (Mercer Consulting, Automobile technologie 2010)

AUTOSAR Basic Software

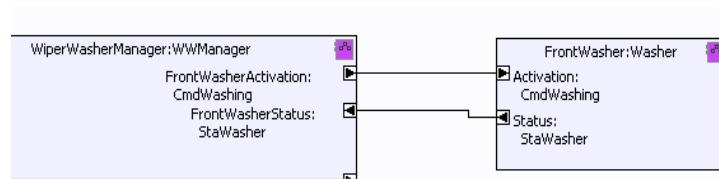


Example: “NVRAM Manager” ensures the storage and maintenance of non-volatile data and is independent of the design of the ECU.



55

Glance on Application Interfaces – Body Domain



- CmdWashing is the interface defined by following information:
 - It is provided by the WiperWasherManager component through the [Washer]Activation port
 - CmdWashing contains one data element command
 - Command is of type t_onoff
 - t_onoff is a RecordType, which describes a generic on/off information

56

AUTOSAR Metamodel**Formal description of all methodology related information**

- The metamodel is modeled in UML
- The structure of the information can be clearly visualized
- The consistency of the information is guaranteed
- Using XML, a data exchange format can be generated automatically out of the metamodel

