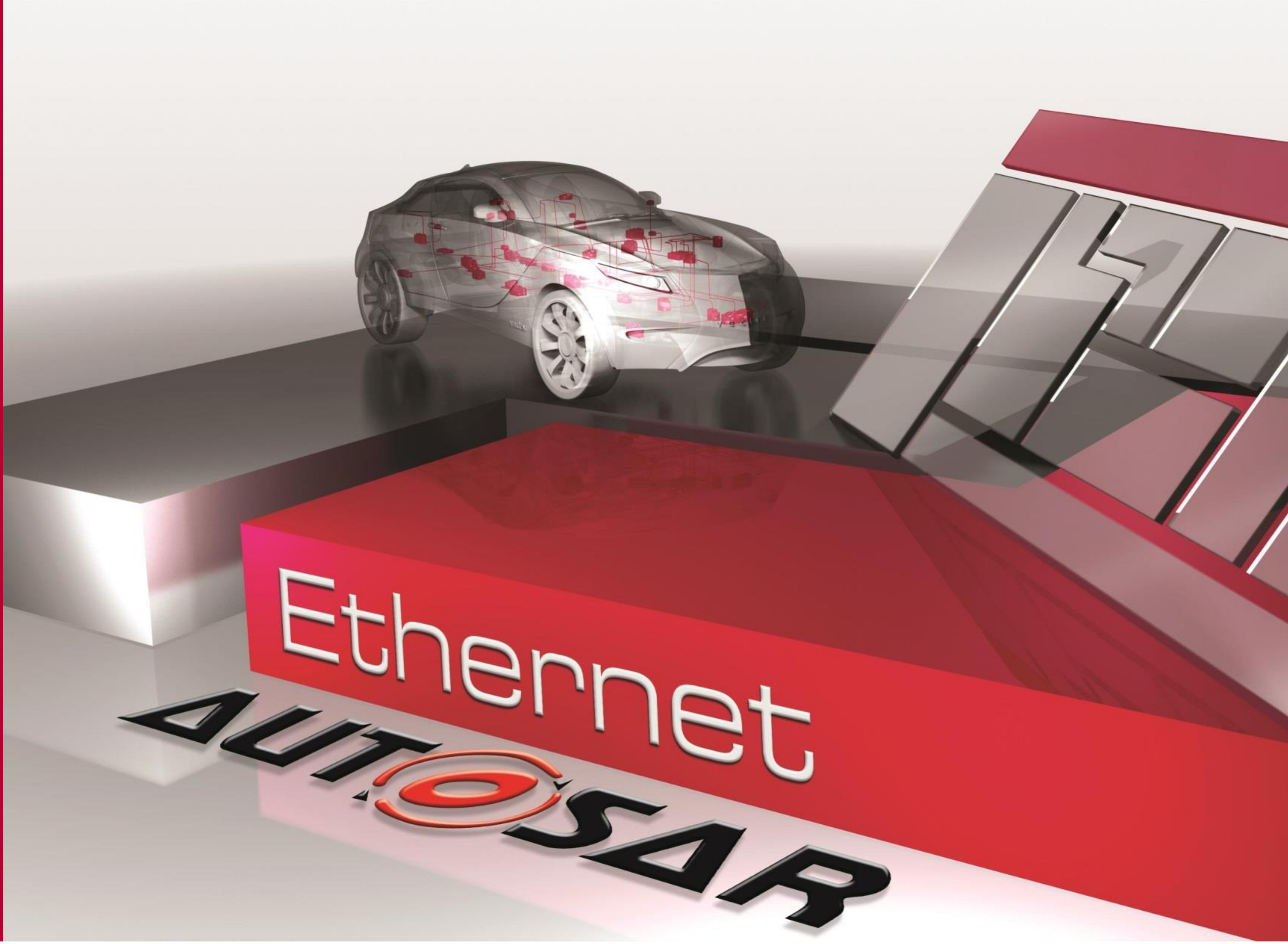


PES Solution Ethernet

The Vector Embedded Ethernet-Stack and its Use-Cases

PES Solution Ethernet Overview

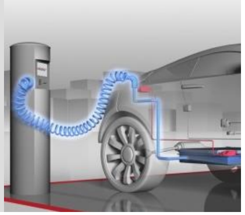
The **PES Solution Ethernet** comprises:

▶▶ Ethernet – MICROSAR.ETH

- Service Discovery, SOME/IP
- UDP Network Management
- Signal- and PDU-based communication
- Diagnostics over Internet Protocol, Flashbootloader
- XCP on Ethernet, XCP Routing
- Mirroring

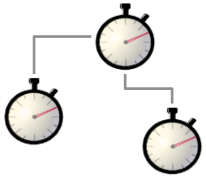
▶▶ Vehicle-to-Grid – MICROSAR.V2G

- Smart Charge Communication
- Customer-specific functions



▶▶ Audio/Video Bridging – MICROSAR.AVB

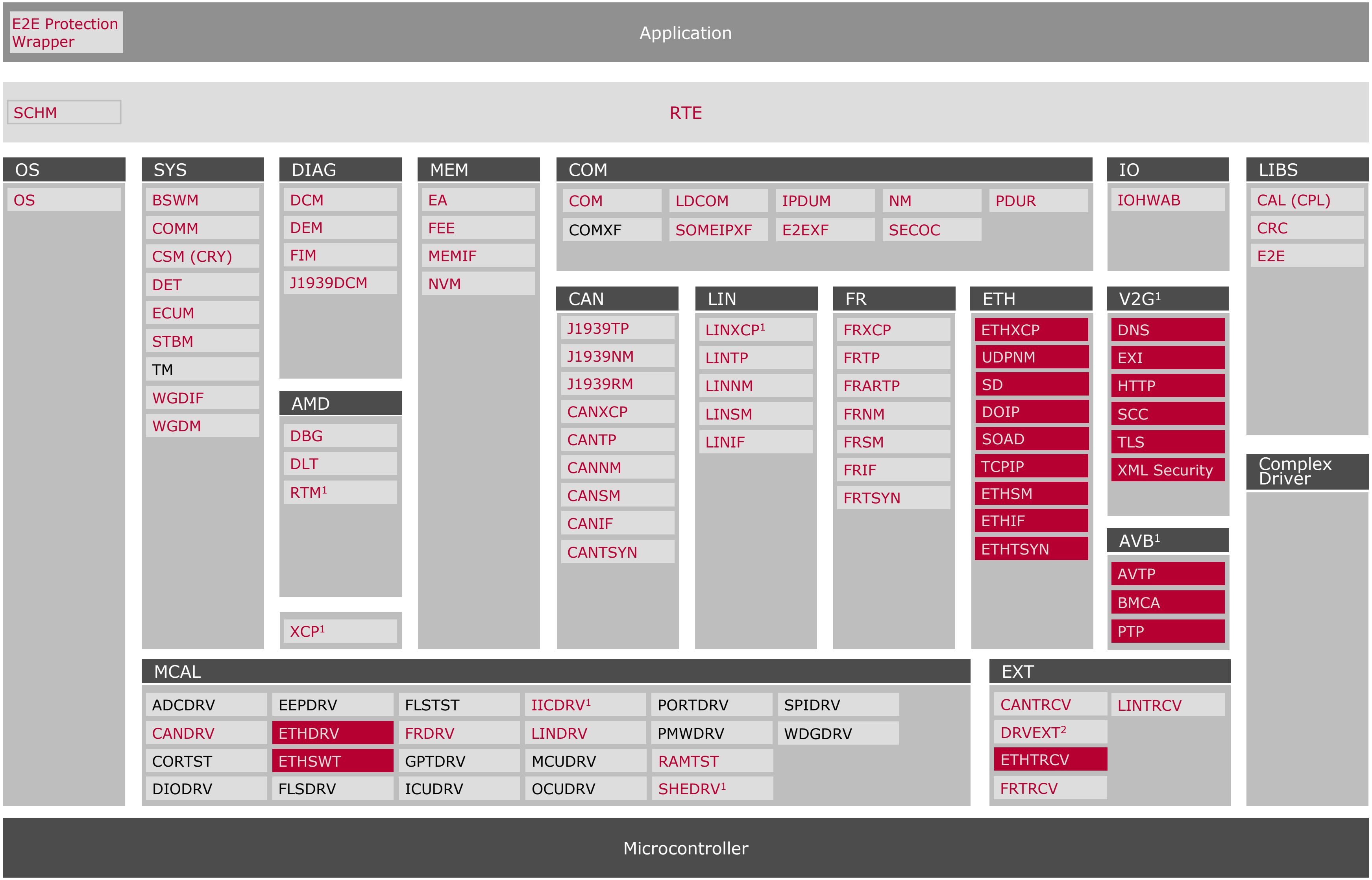
- Audio/Video Transport Protocol
- Generalized Precision Time Protocol / Best Master Clock Algorithm



Agenda

	PES Solution Ethernet Overview	2
>	**AUTOSAR and MICROSAR Software Architecture**	**4**
	MICROSAR.ETH	11
	MICROSAR.V2G	35
	MICROSAR.AVB	38
	Summary	42

AUTOSAR and MICROSAR Software Architecture



E2E Protection Wrapper

Application

SCHM

RTE

OS
- OS

SYS
- BSWM
- COMM
- CSM (CRY)
- DET
- ECUM
- STBM
- TM
- WGDIF
- WGDM

DIAG
- DCM
- DEM
- FIM
- J1939DCM

AMD
- DBG
- DLT
- RTM[1]

XCP[1]

MEM
- EA
- FEE
- MEMIF
- NVM

COM
- COM
- LDCOM
- IPDUM
- NM
- PDUR
- COMXF
- SOMEIPXF
- E2EXF
- SECOC

CAN
- J1939TP
- J1939NM
- J1939RM
- CANXCP
- CANTP
- CANNM
- CANSM
- CANIF
- CANTSYN

LIN
- LINXCP[1]
- LINTP
- LINNM
- LINSM
- LINIF

FR
- FRXCP
- FRTP
- FRARTP
- FRNM
- FRSM
- FRIF
- FRTSYN

ETH
- ETHXCP
- UDPNM
- SD
- DOIP
- SOAD
- TCPIP
- ETHSM
- ETHIF
- ETHTSYN

IO
- IOHWAB

V2G[1]
- DNS
- EXI
- HTTP
- SCC
- TLS
- XML Security

AVB[1]
- AVTP
- BMCA
- PTP

LIBS
- CAL (CPL)
- CRC
- E2E

Complex Driver

MCAL
- ADCDRV
- CANDRV
- CORTST
- DIODRV
- EEPDRV
- ETHDRV
- ETHSWT
- FLSDRV
- FLSTST
- FRDRV
- GPTDRV
- ICUDRV
- IICDRV[1]
- LINDRV
- MCUDRV
- OCUDRV
- PORTDRV
- PMWDRV
- RAMTST
- SHEDRV[1]
- SPIDRV
- WDGDRV

EXT
- CANTRCV
- DRVEXT[2]
- ETHTRCV
- FRTRCV
- LINTRCV

Microcontroller

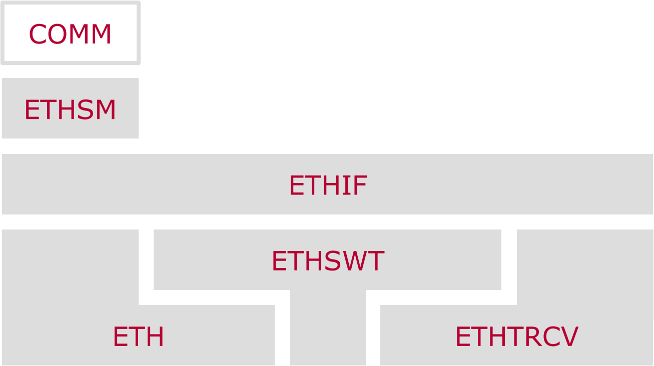
Vector Standard Software

3rd Party Software

[1] Available extensions for AUTOSAR
[2] Includes EXTADC, EEPEXT, FLSEXT, ETHSWTEXT and WDGEXT

AUTOSAR and MICROSAR Software Architecture

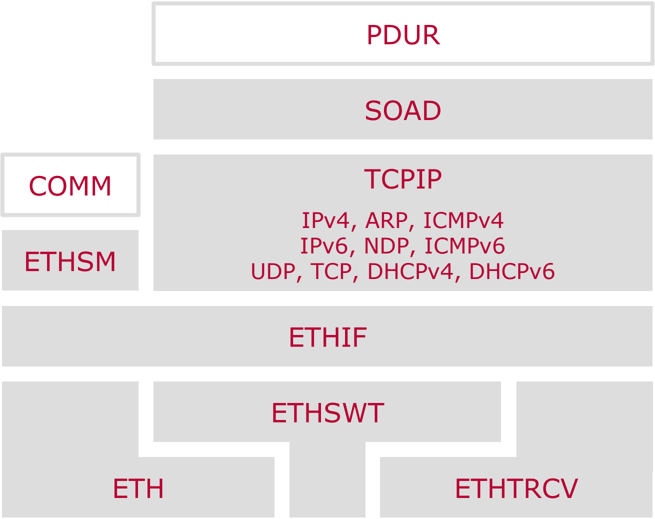
Ethernet Communication Stack



- ETH – Ethernet Driver
  - Uniform API to access all Ethernet controllers of the same type
- ETHTRCV – Ethernet Transceiver Driver
  - Uniform API to access all Ethernet transceivers of the same type
- ETHSWT – Ethernet Switch Driver
  - Configuration of Ethernet switches – e.g. Virtual Local Area Networks (VLANs) and routing tables
  - Uniform API to access all Ethernet switches of the same type (ETHSWTEXT for external switches which are configured e.g. via SPI)
- ETHIF – Ethernet Interface
  - Hardware independent interface to access all Ethernet drivers and Ethernet transceiver drivers
  - Handling of different VLANs
- ETHSM – Ethernet State Manager
  - Enabling and disabling of Ethernet controller and Ethernet transceiver to switch on or off Ethernet communication

AUTOSAR and MICROSAR Software Architecture

Ethernet Communication Stack



- TCPIP – TCP/IP stack
  - IPv4/IPv6 – Internet Protocol version 4/version 6 (IPv4 and IPv6 can be used in parallel on the same ECU)
  - ARP – Address Resolution Protocol (IPv4)
  - NDP – Neighbor Discovery Protocol (IPv6)
  - ICMPv4/v6 – Internet Control Message Protocol
  - UDP – User Datagram Protocol
  - TCP – Transmission Control Protocol
  - DHCPv4/v6 – Dynamic Host Configuration Protocol
    - v4: Client and Server
    - v6: Client only
- SOAD – Socket Adaptor
  - Transformation of socket-based into AUTOSAR PDU-oriented communication

AUTOSAR and MICROSAR Software Architecture

Supported Hardware

- Microcontrollers / Ethernet controllers
  - MPC (FEC and ENET)
  - TriCore, Aurix
  - V850, RH850
  - Jacinto 6 (incl. switch functionality)
  - AR7000 (Powerline communication)
  - CANoeEmu (Simulating an Ethernet controller within a CANoe DLL)
- Ethernet transceivers
  - BCM89810 (BroadR-Reach)
  - DP83848 (100BASE-TX)
  - AR7000 (Powerline communication)
  - CANoeEmu (Simulating an Ethernet transceiver within a CANoe DLL)
  - Generic (Support of all transceivers which have a MII interface)
- Ethernet switches
  - BCM89501 (BroadR-Reach)

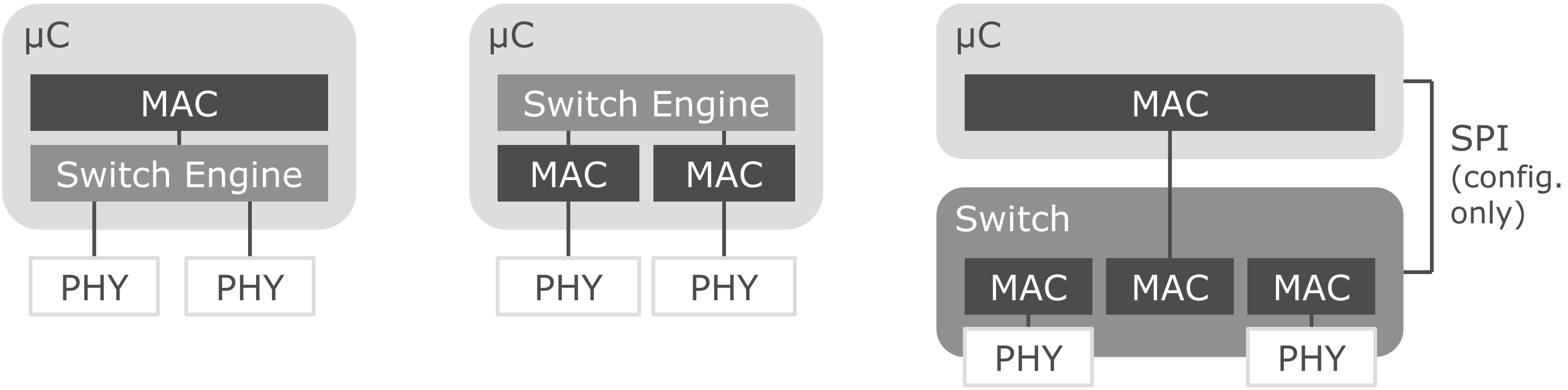
Further controllers, transceivers and switches can be supported on request!

AUTOSAR and MICROSAR Software Architecture

Supported Hardware

Ethernet Switches

- Automotive Ethernet switches offer different configuration possibilities
  - VLAN
  - Forwarding tables
  - Queuing mechanisms
  - …
- AUTOSAR architecture fits to all known Ethernet switch architectures



Agenda

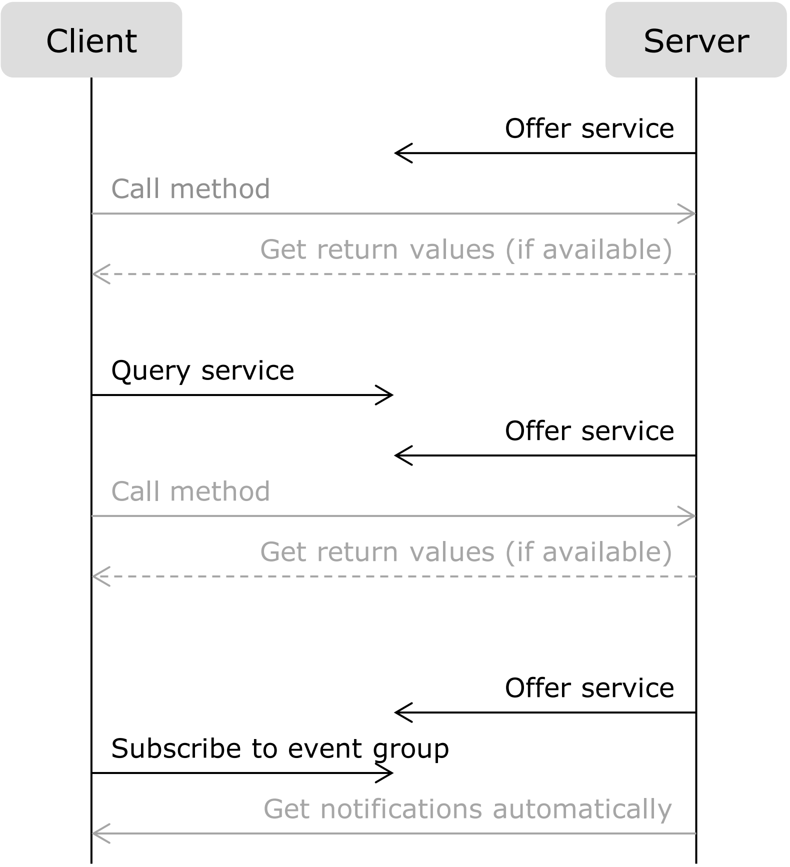
	PES Solution Ethernet Overview	2
	AUTOSAR and MICROSAR Software Architecture	4
>	**MICROSAR.ETH**	**11**
	MICROSAR.V2G	35
	MICROSAR.AVB	38
	Summary	42

Service Discovery (SD)

- Service-oriented communication scheme instead of a classical signal-oriented approach
- What is a “Service”?
  - A Service can contain “Methods” which can be called by other ECUs, this mechanism is known as Remote Procedure Call (RPC)
  - A Service can contain “Events” to which other ECUs can subscribe to be informed about changes or updates
  - There are Service providers (servers) and Service consumers (clients)
- What is the purpose of Service Discovery?
  - A provider announces the availability and implicitly the location of a Service via Service Discovery to other ECUs
  - A consumer knows the availability of a Service, can call Methods and can subscribe to offered Event groups
- Service Discovery was first specified in AUTOSAR 4.1.1

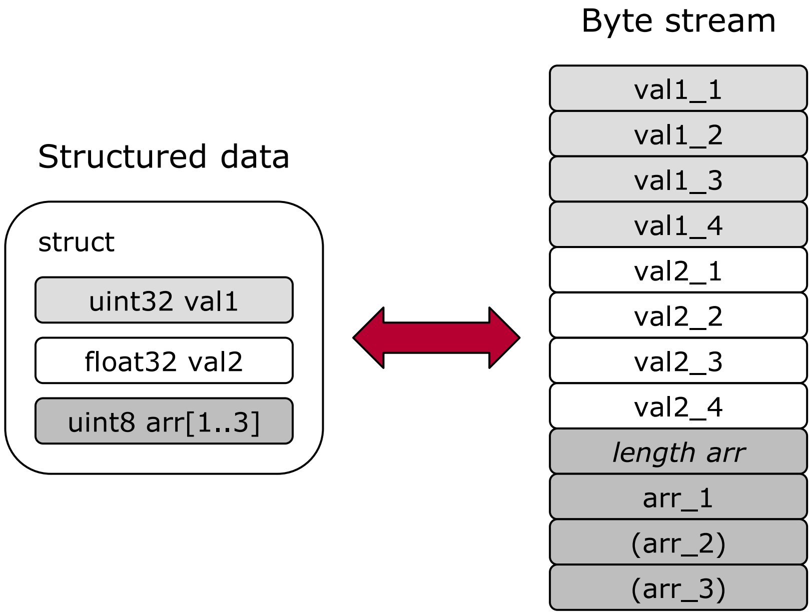
In-Vehicle Ethernet Communication

- Advantages
  - Avoid sending of invalid signals
    - Save bandwidth
  - Avoid sending multi- and broadcasts but use unicast
    - Save bandwidth on alternative communication paths (use advantage to Ethernet as a switched network)
  - Reduce CPU load
  - Dynamic relocation of Services possible
- Modes
  - Announce on startup
  - Query/Announce
  - Publish/Subscribe



Scalable Service-Oriented Middleware over IP (SOME/IP)

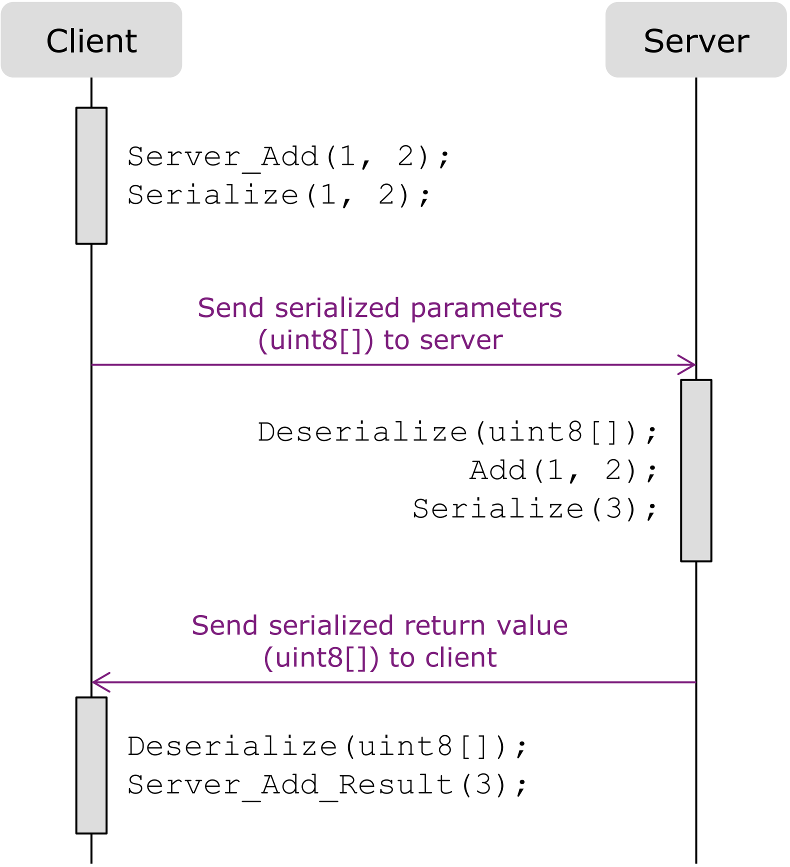
- SOME/IP is an automotive serialization protocol
  - Definition of a header format which supports RPCs, i.e. calling a function on a remote server ECU like it would be executed on the own ECU
  - Definition how application data shall be serialized to the on-the-wire payload
    - Support of basic data types, complex data types (e.g. C-struct), static and dynamic array data types
    - Independent on endianness
  - Designed for AUTOSAR and non-AUTOSAR ECUs



MICROSAR.ETH

In-Vehicle Ethernet Communication

- SOME/IP does not require Service Discovery and vice versa; however, they are designed to work together
  - Methods calls can be performed via SOME/IP messages
  - Event notifications can be sent via SOME/IP messages
  - Service Discovery messages like ‘Offer’ and ‘Query’ use the SOME/IP header
- SOME/IP was specified by BMW
  - Auxiliary document in AUTOSAR 4.1.1
  - SOME/IP will be specified in AUTOSAR 4.2.1



Client

Server

```
Server_Add(1, 2);
Serialize(1, 2);
```

Send serialized parameters (uint8[]) to server

```
Deserialize(uint8[]);
         Add(1, 2);
       Serialize(3);
```

Send serialized return value (uint8[]) to client

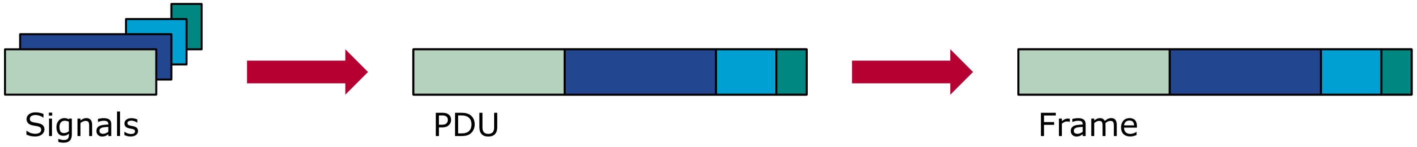
```
Deserialize(uint8[]);
Server_Add_Result(3);
```

UDP Network Management (UDPNM)

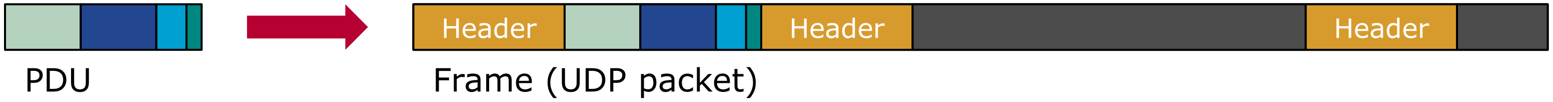
- Coordination of the transition between normal operation and bus-sleep mode of an Ethernet network
  - Periodic broadcast messages are sent by nodes which want to keep the NM-cluster awake
  - No master node
  - Node detection (detect all present nodes in a network)
  - Ready sleep detection (detect if all nodes in a network are ready for bus-sleep mode)
  - Partial Networking support added with AUTOSAR 4.1.1
  - Similar to network management on CAN
- No wake-up based on Ethernet communication possible
  - Additional bus connection or wake-up line necessary
  - Transceiver support missing
    - Power consumption in sleep mode is too high (all switches have to be powered up)
    - New transceivers will be available soon (2014)

Signal-/PDU-based communication

- Mapping of Signals onto a PDU and a PDU onto a frame
  - Similar to classical communication on CAN

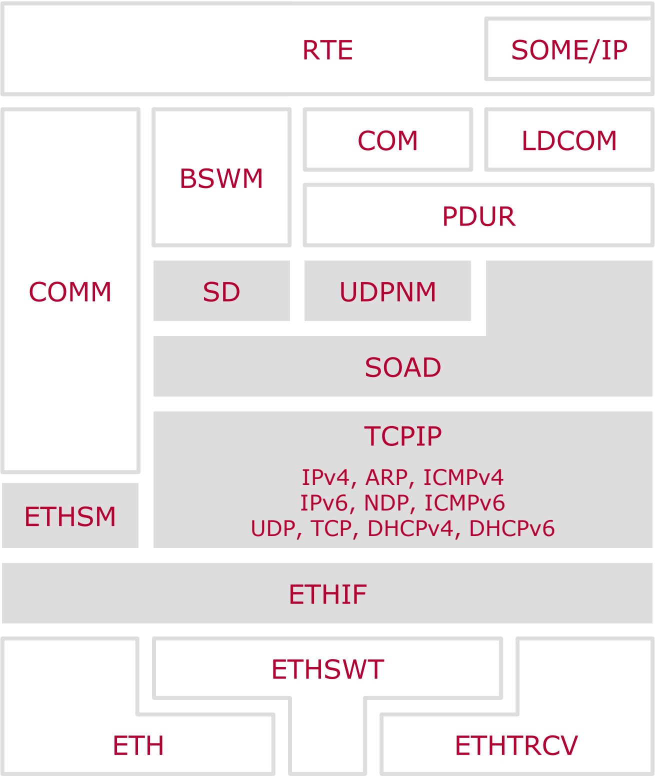


- Transmission and reception of multiple PDUs in one Ethernet frame to save resources
  - Fan-out mechanism allows to transmit one PDU to several destinations via unicast
  - Location of a PDU within a frame is dynamic
    - Socket Adaptor adds and removes a small header to differentiate between PDUs which comprises a PDU identifier and a length field
    - Different frame triggering mechanisms available



MICROSAR.ETH

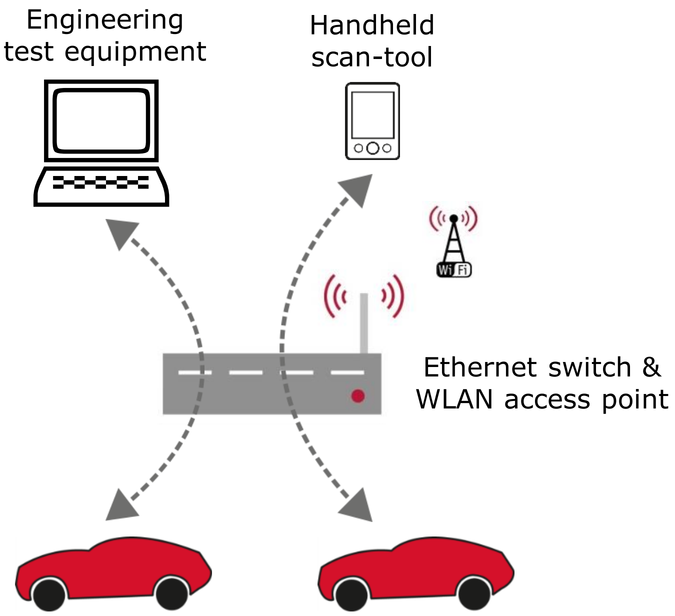
In-Vehicle Ethernet Communication



- SD – Service Discovery
  - > Management of service states
  - > Configuring communication paths over the Socket Adaptor e.g. disable routing if service is not available
- SOME/IP – Scalable Service-Oriented Middleware over Internet Protocol
  - > Serialization of application data and deserialization of received data
  - > Since AUTOSAR 4.2.1 specified as RTE transformer
  - > Large Data COM (LDCOM) can be used to save resources
- UDPNM – UDP Network Management
  - > Coordinated shut down of Ethernet ECUs
  - > Connection to NMIF is not shown in the figure

Diagnostics over Internet Protocol (DoIP)

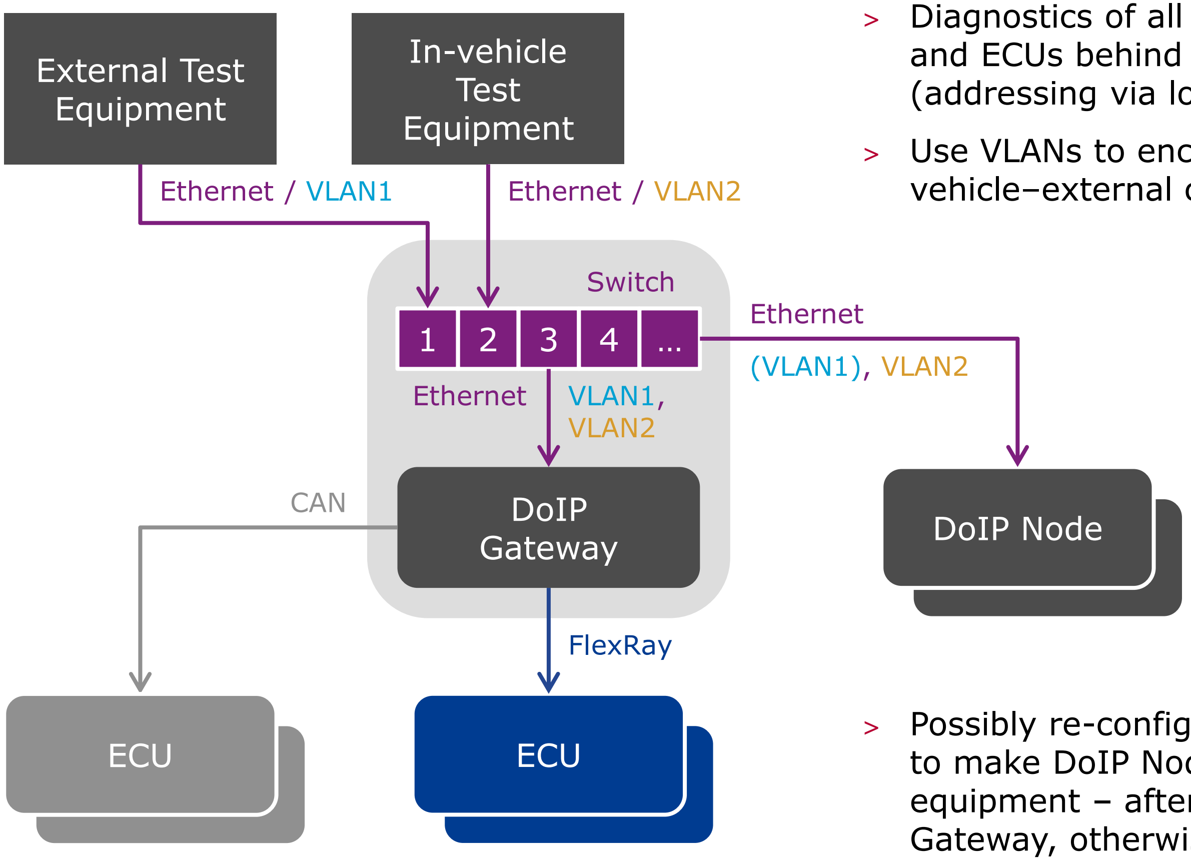
- Use-cases
  - Diagnostics (repair shop, legislator)
  - ECU re-programming
    Manufacturer: End of line programming
    Repair shop: Software update
- Advantages
  - High-speed access to vehicle ECUs
  - Gateway to existing bus systems like CAN and FlexRay
  - Parallel flash download
  - Ethernet, TCP/IP as well-known technology
  - New possibility for production and repair shop infrastructure (WLAN)


Engineering test equipment
Handheld scan-tool
Wi Fi
Ethernet switch & WLAN access point

DoIP Diag. Message	Eth	IP	TCP	DoIP	UDS	...	...	Eth

MICROSAR.ETH

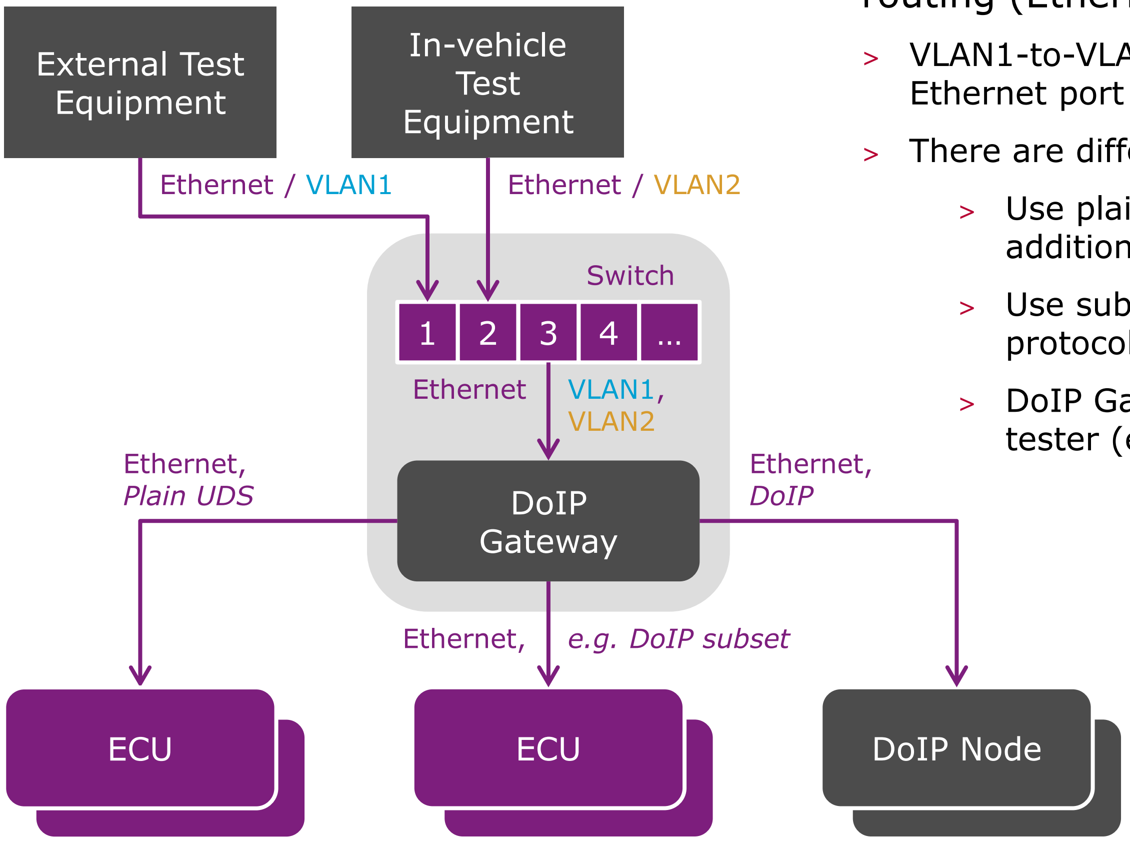
Diagnostics, Measurement and Calibration



- ISO 13400
  - Diagnostics of all DoIP Nodes, DoIP Gateways and ECUs behind the DoIP Gateways (addressing via logical DoIP address)
  - Use VLANs to encapsulate vehicle-internal from vehicle–external communication
  - Possibly re-configure the switch during runtime to make DoIP Nodes visible to the external test equipment – after authentication to the DoIP Gateway, otherwise the DoIP Gateway has to route messages

MICROSAR.ETH

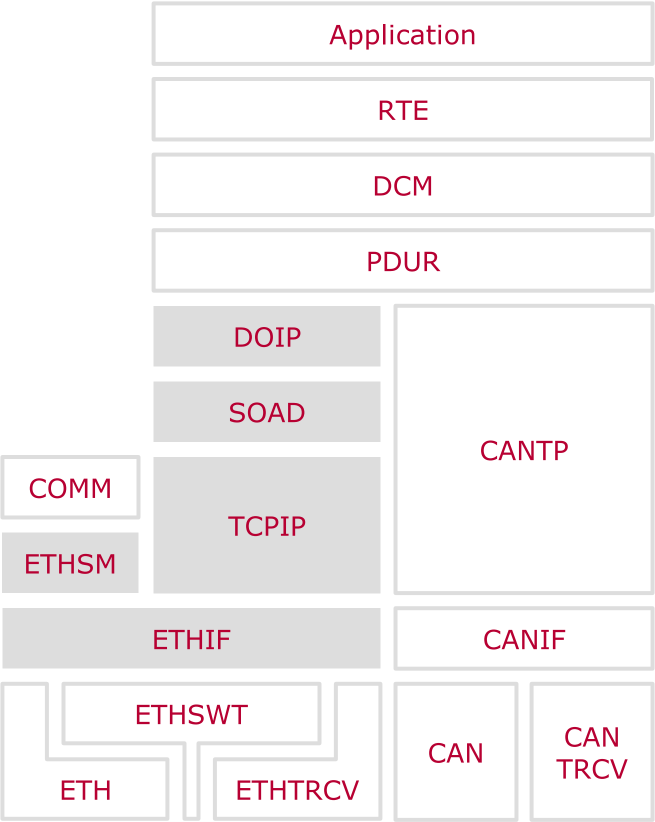
Diagnostics, Measurement and Calibration



- ISO 13400 does not specify DoIP-to-DoIP routing (Ethernet behind the gateway)
  - VLAN1-to-VLAN2 conversion on the same Ethernet port is also considered as routing
  - There are different possible solutions
    - Use plain UDS over TCP/IP without additional protocol
    - Use subset of DoIP or other OEM-specific protocol
    - DoIP Gateway implements a diagnostic tester (emulation)

MICROSAR.ETH

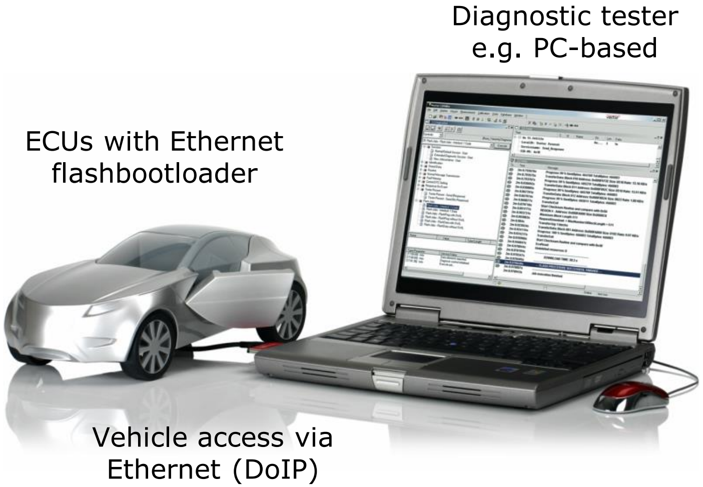
Diagnostics, Measurement and Calibration



- DOIP – Diagnostics over Internet Protocol
  - Implementation of ISO 13400-2
  - The handling of the DoIP protocol was specified as Socket Adaptor plug-in in AUTOSAR 4.0.x and became a separate module in AUTOSAR 4.1.x
- PDUR – PDU Router
  - Central module for PDU forwarding and routing e.g. DoIP <-> Dcm and DoIP <-> CanTp
- DCM – Diagnostic Communication Manager
  - Implementation of the diagnostic protocol (UDS)
  - Interaction with the Diagnostic Event Manager (DEM) to get diagnostic event information
  - Interaction with the application over the RTE to query diagnostic information and execute procedures

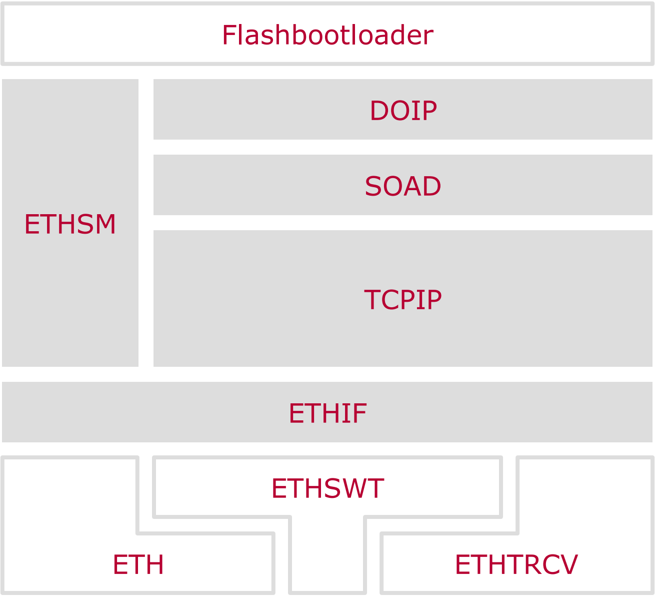
Ethernet Flashbootloader (FBL)

- Fast ECU (re-)programming over Ethernet via DoIP
  - 100MBit/s instead of max. 1MBit/s with CAN
- Use-cases
  - End-of-line (EOL) ECU programming
  - After-sales ECU re-programming
  - ECU development
- Properties
  - Independent application which remains permanently in the ECU
  - The FBL is based on the MICROSAR.ETH BSW modules, but the source code cannot be shared between FBL and normal application
  - Because of the special flash driver, the FBL is a hardware dependent application
- Offered as additional service



MICROSAR.ETH

Diagnostics, Measurement and Calibration



- Flashbootloader
  - Communication (COMM and PDUR) wrapper
  - Flash/EEPROM driver
  - FBL security module
  - FBL diagnostics
  - FBL application

Universal (X) Measurement and Calibration Protocol (XCP)

- Successor of CCP – CAN Calibration Protocol
- Used for measurement, calibration, bypassing and ECU re-programming
- The XCP protocol is split into transport and protocol layer
  - The protocol layer is identical for all network technologies i.e. CAN, FlexRay, Ethernet, USB
  - The transport layer is specific for each network technology (different header and trailer data)
- ASAM AE MCD-1 XCP V1.1.0
  - ASAM AE MCD-1 XCP V1.2.0 only adds CAN-FD as new transport layer, no functional changes

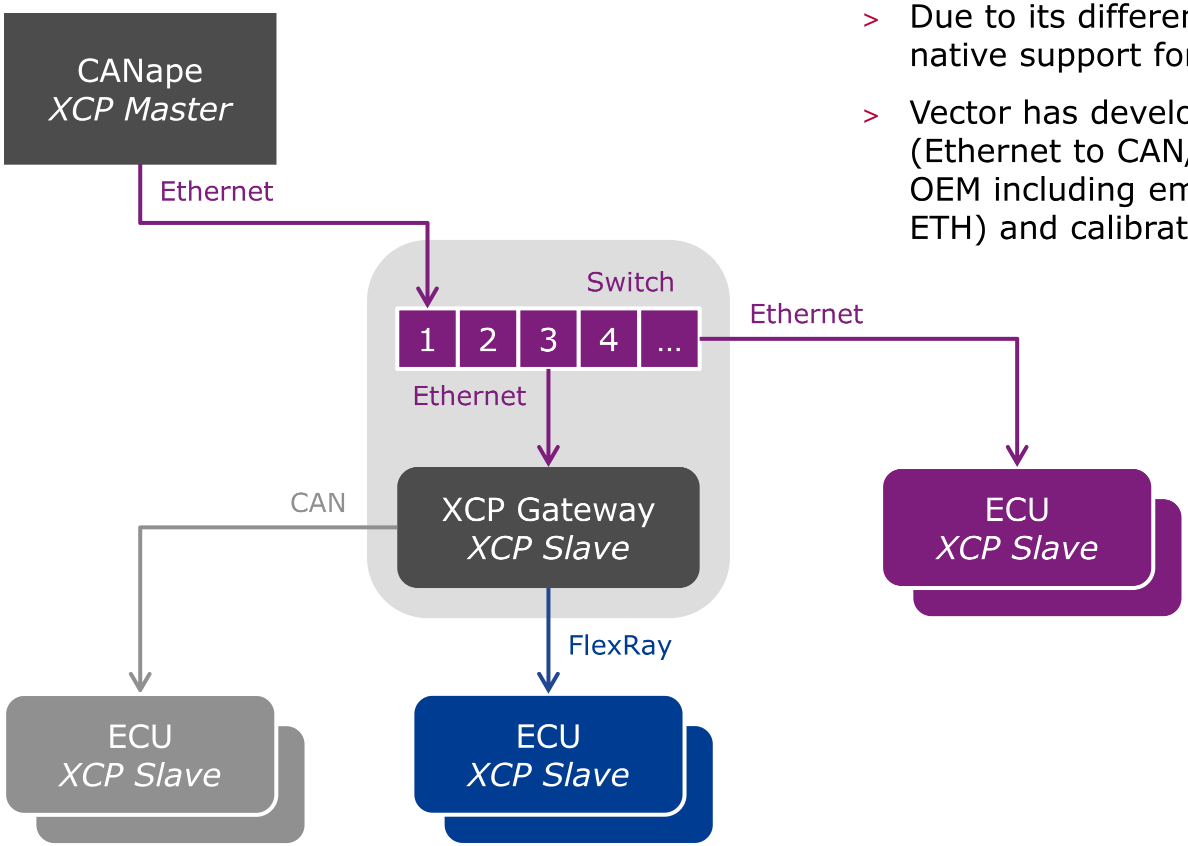
XCP on Ethernet	Eth	IP	UDP/TCP	XCP H.	XCP P.	…	…	Eth

XCP H. = XCP Header (transport layer specific)

XCP P. = XCP Packet (protocol layer)

MICROSAR.ETH

Diagnostics, Measurement and Calibration

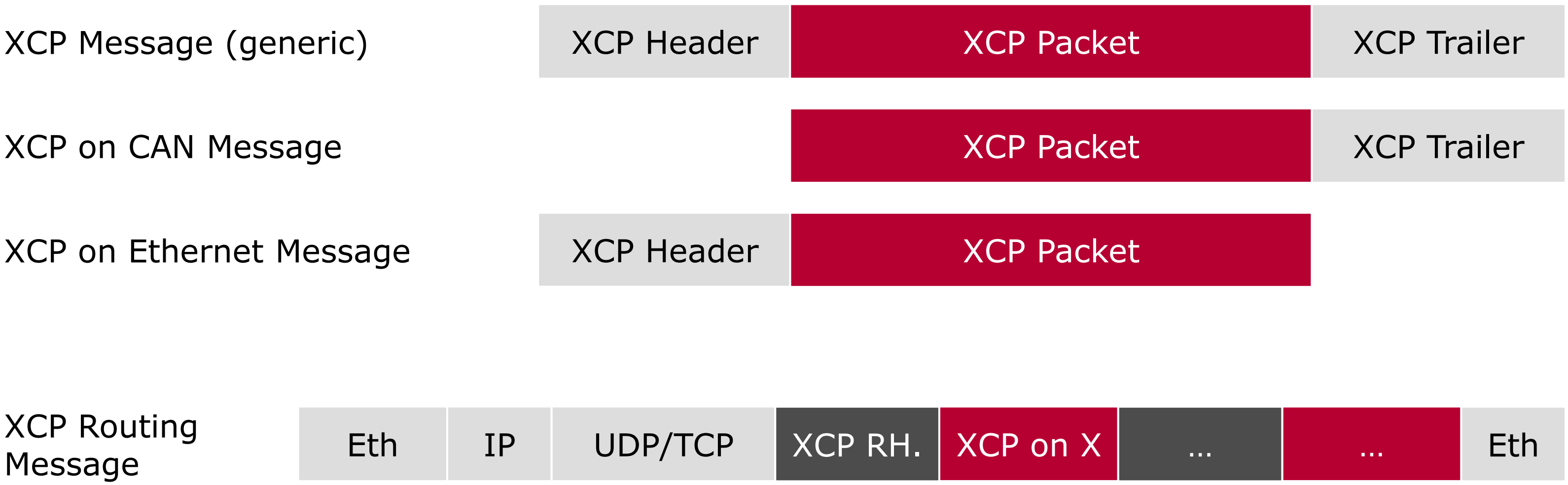


- XCP Routing
  - Due to its different transport layers, XCP has no native support for routing
  - Vector has developed a solution for XCP routing (Ethernet to CAN/FlexRay) together with an OEM including embedded software (MICROSAR ETH) and calibration tool (CANape)

MICROSAR.ETH

Diagnostics, Measurement and Calibration

- XCP Routing
  - Route XCP data from a XCP master over a gateway to multiple ECUs and back
    - Send XCP on CAN/FlexRay messages over Ethernet from a XCP master to the gateway
    - UDP or TCP can be used as transport protocol
    - Cascading of XCP on X messages possible

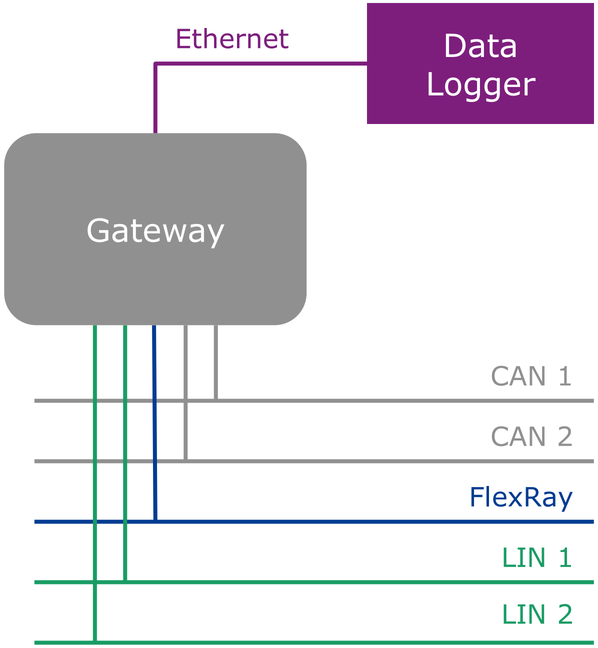


XCP RH. = XCP Routing Header (Vector-specific extension)

XCP on X = XCP on CAN/FlexRay message

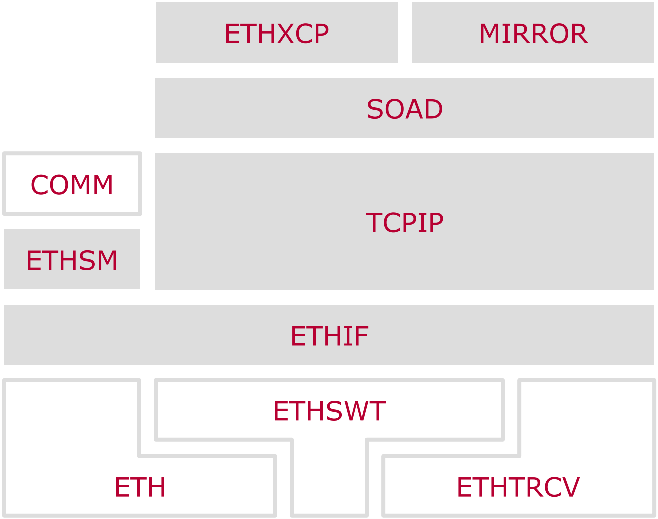
Mirroring

- How can vehicle internal bus traffic be logged and analyzed if Ethernet is the only access point to the vehicle and without additional logging equipment?
- Solution: Mirroring of CAN/LIN/FlexRay bus traffic to Ethernet
  - Mirroring of complete communication
  - Simultaneous mirroring of all connected busses possible
  - Multiple CAN/LIN/FlexRay messages are packed in one Ethernet packet
  - Additional information is added to the mirrored messages
  - Mirror functionality can be switched on/off e.g. by a diagnostic command
  - Implementation as CDD above the Socket Adaptor with direct access to bus drivers
- Analysis of mirrored data in CANoe



MICROSAR.ETH

Diagnostics, Measurement and Calibration



- ETHXCP – XCP on Ethernet
  - Ethernet specific transport layer of
    ASAM AE MCD-1 XCP V1.1.0
- SOAD – Socket Adaptor
  - Vector specific extensions to the AUTOSAR specification
    to support XCP routing and Mirroring
- MIRROR – Mirroring
  - Complex Driver above the Socket Adaptor with direct access to communication drivers

Agenda

PES Solution Ethernet Overview 2

AUTOSAR and MICROSAR Software Architecture 4

MICROSAR.ETH 11

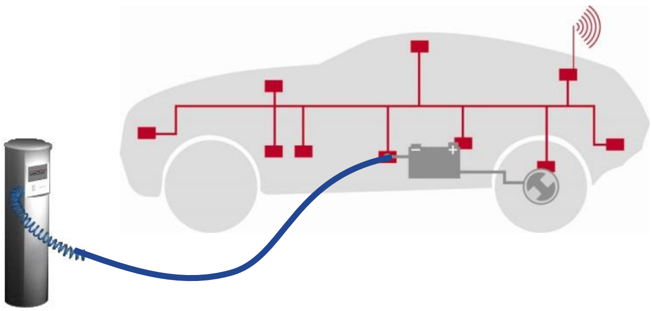
> **MICROSAR.V2G** **35**

MICROSAR.AVB 38

Summary 42

MICROSAR.V2G

Smart Charge Communication (SCC)

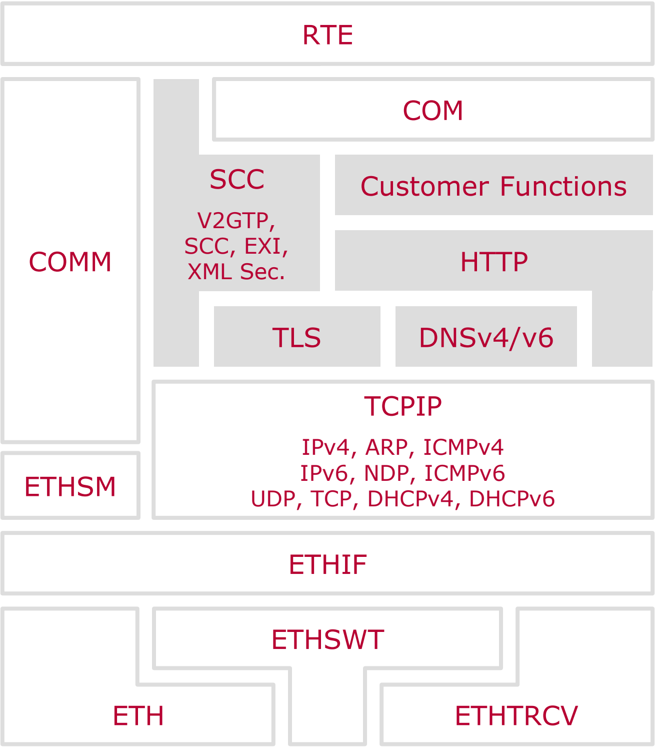


- When and how to charge an electric vehicle?
  - How much energy is available/required?
  - How to pay?
- ISO 15118
  - AC and DC charging
  - Profile: Plug and Charge (PnC) – charging in a public environment with billing
  - Profile: External Identification Means (EIM) – “simple” charging
  - Value Added Services (VAS)
- DIN 70121
  - DC charging (similar to ISO DC EIM – no encryption, no payment)

Customer Specific Functions

- If the vehicle is connected to the Internet e.g. via a charge spot, additional web-services may be available or implemented by using e.g. HTTP

MICROSAR.V2G



- TLS – Transport Layer Security
  - Encryption and decryption of TCP streams
- SCC – Smart Charge Communication
  - Implementation of the V2GTP (Vehicle-to-Grid transport protocol)
  - Efficient XML Interchange (EXI) and XML Security are implemented in separate modules
- DNSv4/v6 – Domain Name Service
  - Resolver only: Resolution of URLs into IP addresses (IPv4/IPv6)
- HTTP – Hypertext Transfer Protocol
  - String-based communication with a server
- Customer Functions
  - Customer specific functions, which e.g. are based on HTTP communication
  - An implementation of XML Engine and JSON (JavaScript Object Notation) is also available

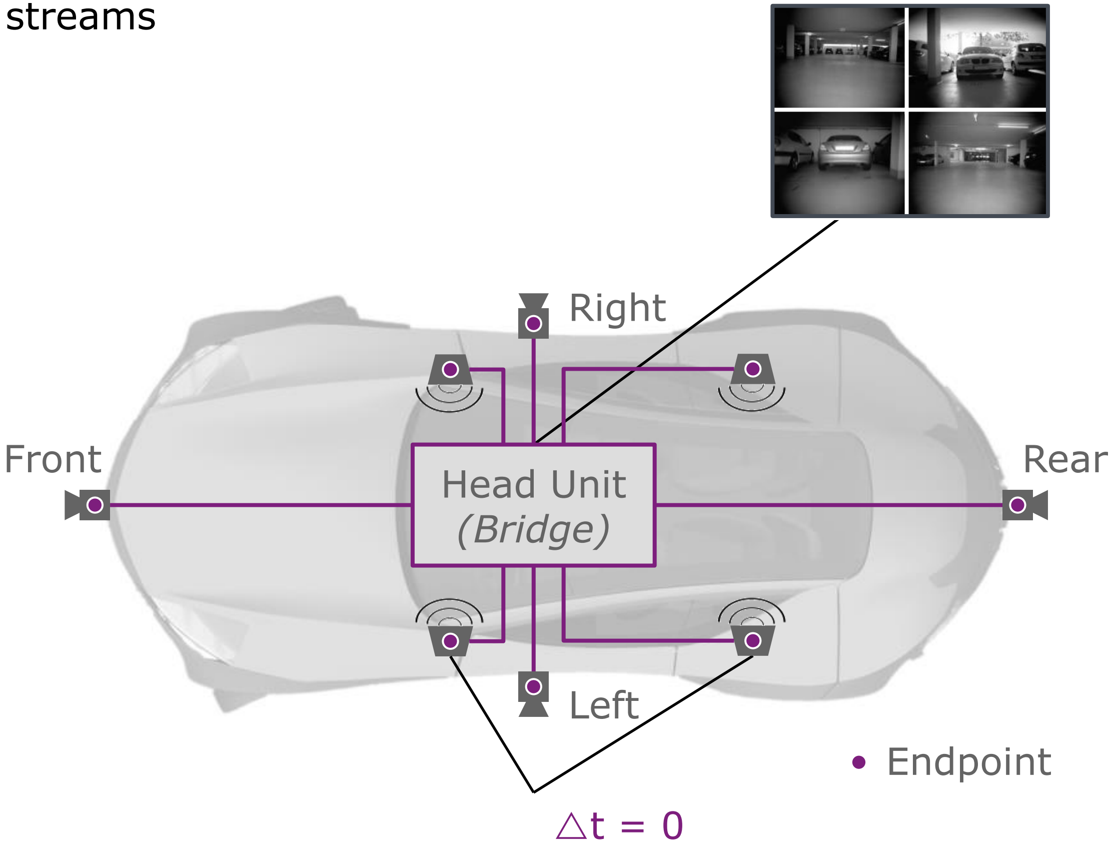
Agenda

	PES Solution Ethernet Overview	2
	AUTOSAR and MICROSAR Software Architecture	4
	MICROSAR.ETH	11
	MICROSAR.V2G	35
>	**MICROSAR.AVB**	**38**
	Summary	42

MICROSAR.AVB

Audio/Video Bridging (AVB)

- Transport audio and video streams
  - Through standard Ethernet network technology
  - With simple cabling
  - Fast and in real-time
  - Well synchronized and prioritized

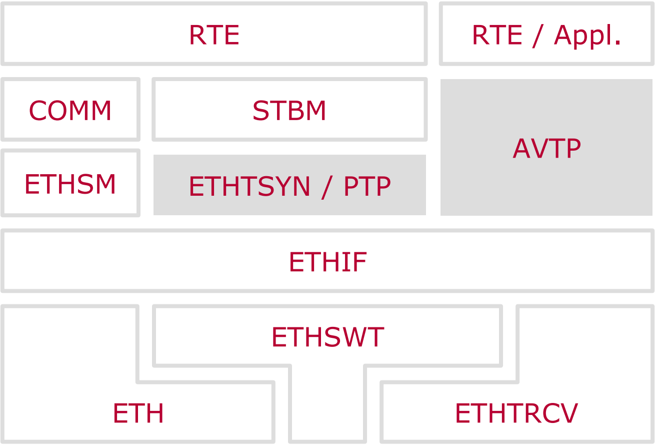


MICROSAR.AVB

- Important IEEE specifications for bridging & management
  - Introduction and Overview
    - IEEE 802.1BA – Audio Video Bridging (AVB) Systems
  - Generalized Precision Time Protocol (gPTP) and Best Master Clock Algorithm (BMCA)
    - IEEE 802.1AS – Timing and Synchronization for Time-Sensitive Applications
    - References IEEE 1588
  - Traffic shaping
    - IEEE 802.1Qav – Forwarding and Queuing Enhancements for Time-Sensitive Streams
  - Stream management
    - IEEE 802.1Qat – Stream Reservation Protocol (SRP)
    - Dynamic stream announcement with admission control
    - Static implementation for automotive possible

- Audio/Video data transmission and reception
  - Audio/Video Transport Protocol (AVTP)
    - IEEE 1722(a) – Layer 2 Transport Protocol for Time Sensitive Applications
(a): Automotive version in draft status – covers encryption, simple A/V streams and formats, automotive message types within an A/V stream

MICROSAR.AVB



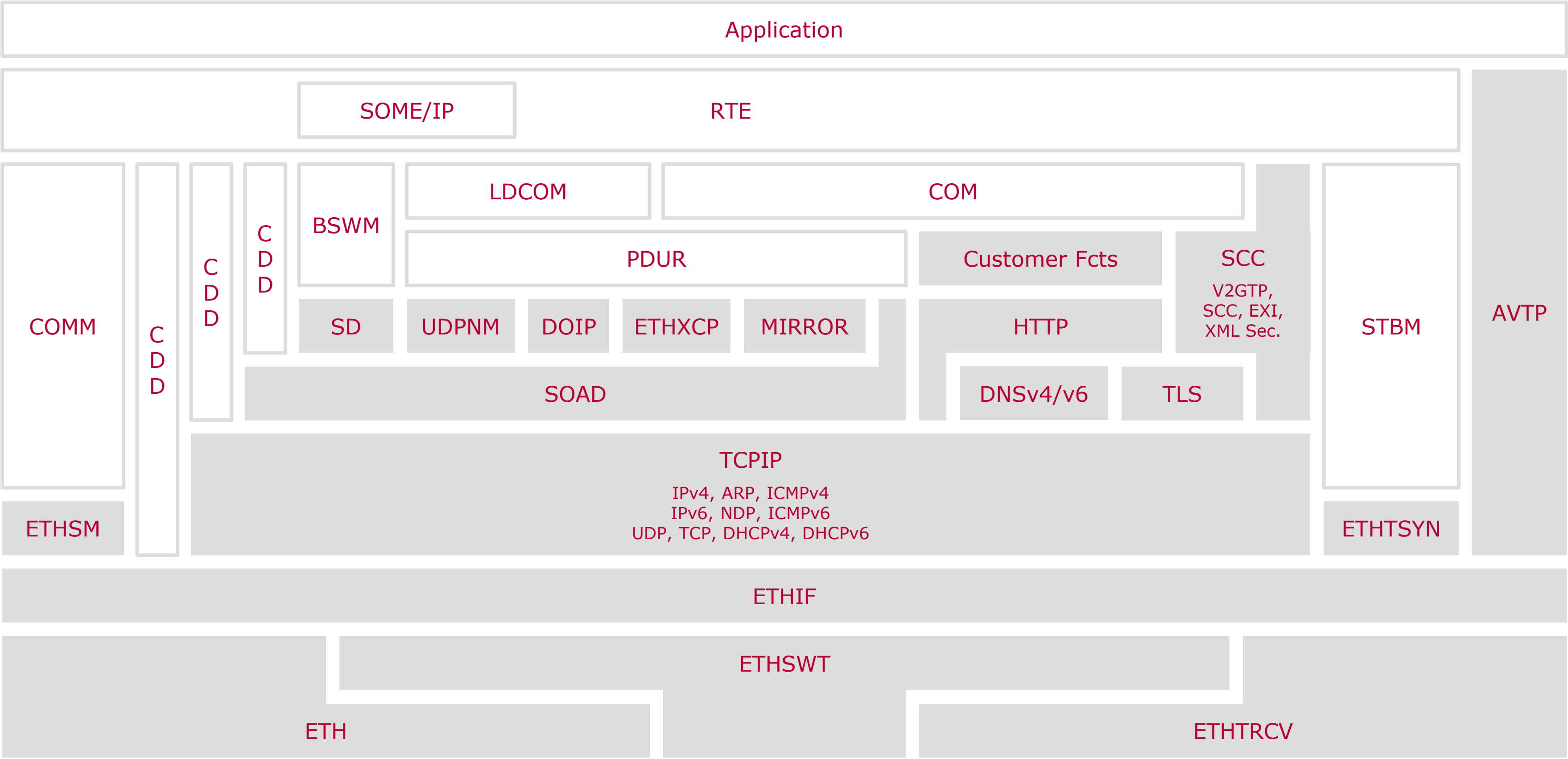
- ETHTSYN – Ethernet Time Synchronization
  - Time synchronization between Ethernet ECUs
  - The Synchronized Time Base Manager (STBM) acts as coordinator between different networks
    - ETHTSYN can also be used without STBM
  - Implements the Generalized Precision Time Protocol (gPTP)
    - IEEE 802.1AS
  - Is specified by AUTOSAR 4.2
  - Is part of MICROSAR.ETH but because it implements gPTP, it is explained in context of MICROSAR.AVB
  - Best Master Clock Algorithm (BMCA) to determine the best clock in the system
    - Can be provided on request
- AVTP – Audio/Video Transport Protocol
  - Transmission and reception of audio and video streams
  - IEEE 1722(a)

Agenda

PES Solution Ethernet Overview	2
AUTOSAR and MICROSAR Software Architecture	4
MICROSAR.ETH	11
MICROSAR.V2G	35
MICROSAR.AVB	38
> Summary	**42**

Summary

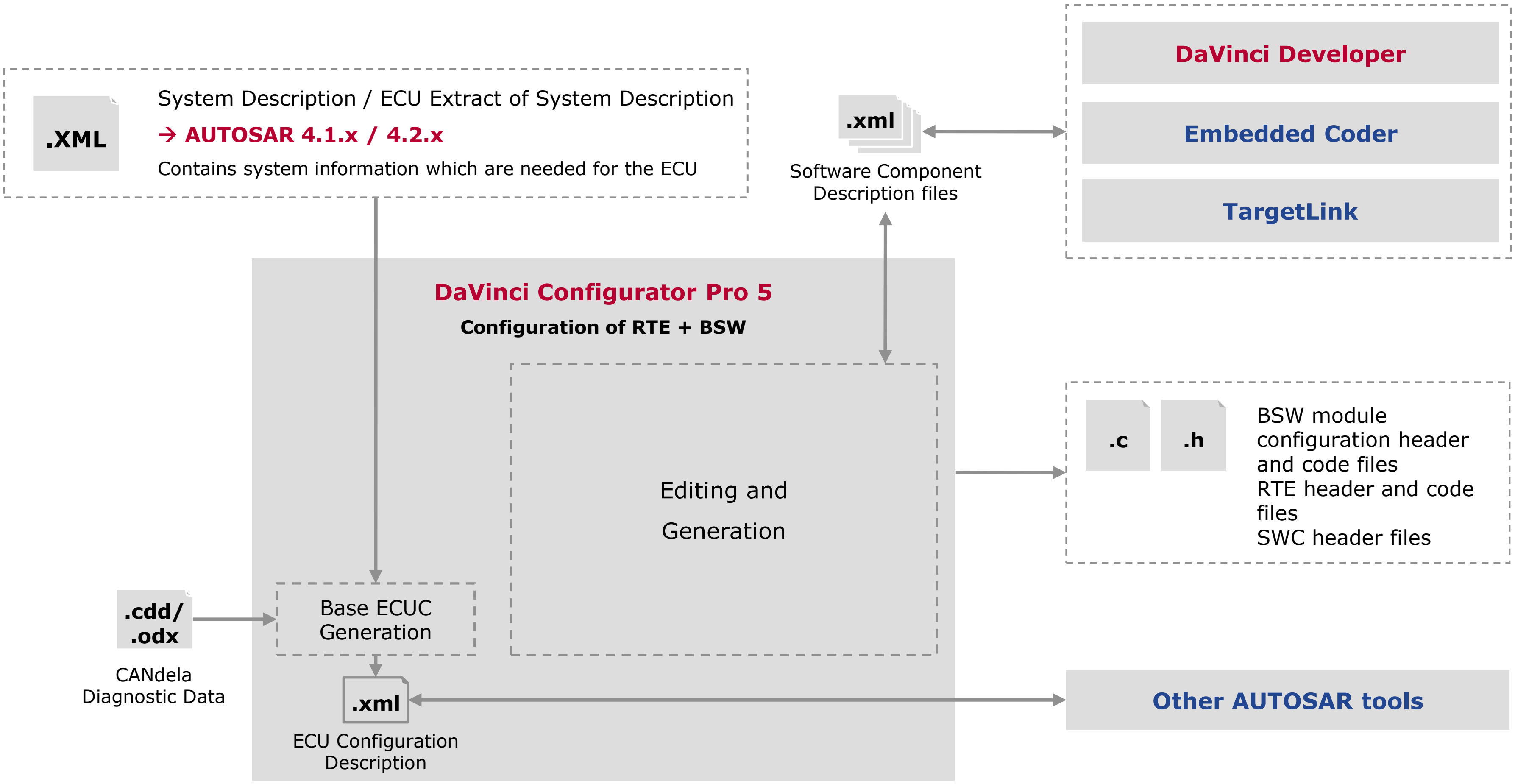
Overall MICROSAR Ethernet Architecture



- MICROSAR.ETH is the basis of MICROSAR.V2G and MICROSAR.AVB
  - All clusters and use-cases can run in parallel
  - Figure does not show ETHSWTEXT and the flashbootloader

Summary

ECU Configuration Flow for MICROSAR.ETH



.XML

System Description / ECU Extract of System Description

→ **AUTOSAR 4.1.x / 4.2.x**

Contains system information which are needed for the ECU

.xml

Software Component Description files

DaVinci Developer

Embedded Coder

TargetLink

DaVinci Configurator Pro 5

Configuration of RTE + BSW

Editing and Generation

.c .h

BSW module configuration header and code files
RTE header and code files
SWC header files

.cdd/ .odx

CANdela Diagnostic Data

Base ECUC Generation

.xml

ECU Configuration Description

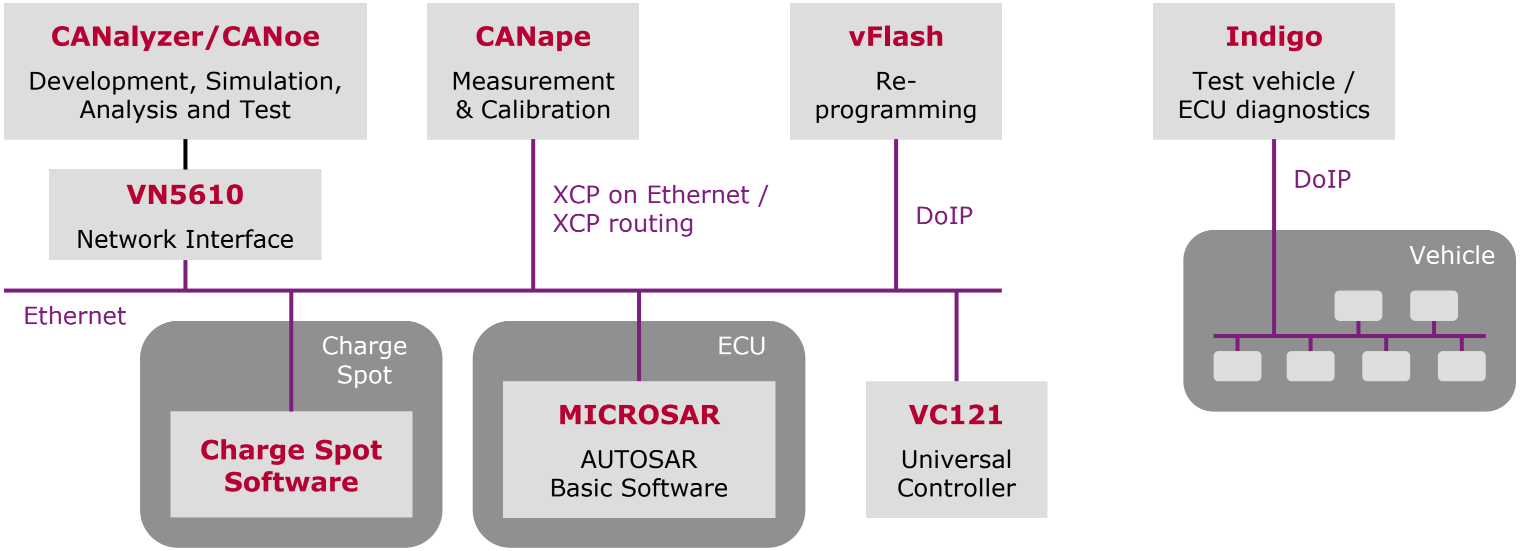
Other AUTOSAR tools

Summary

Vector Ethernet Solution

Ethernet and TCP/IP in Vehicles

- Ethernet and TCP/IP technologies have made their way into the vehicle and they are expected to extend into new application areas
- Automotive OEMs and suppliers are facing diverse challenges here
- Vector supports you in meeting these challenges with professional tools, embedded software and services



Thank you for your attention.

For detailed information about Vector
and our products please visit

www.vector.com

Author:

Weber, Marc

Vector Informatik GmbH