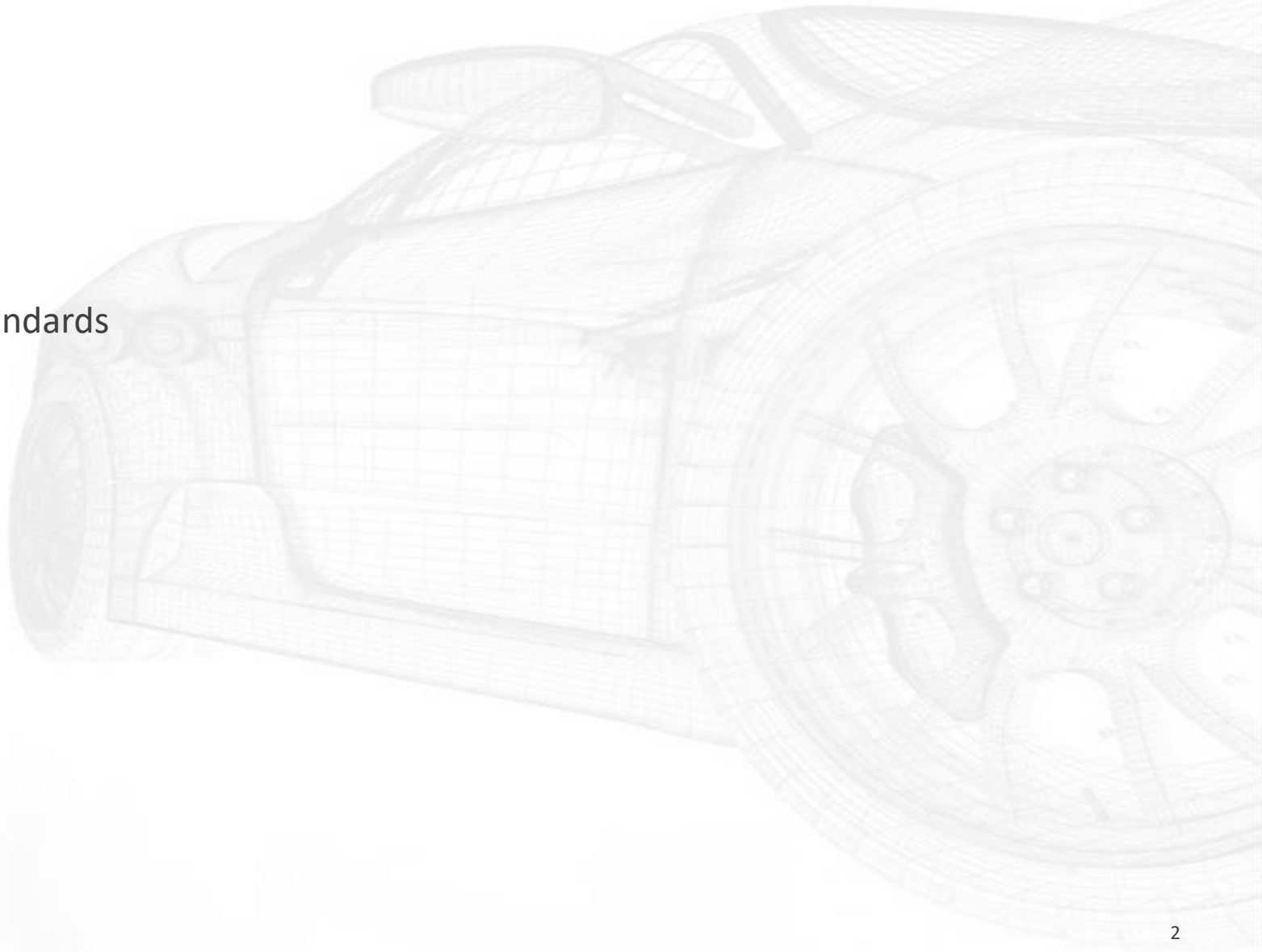
EB tresos Classic AUTOSAR Training
Introduction & Basic AUTOSAR Concepts

Elektrobit

Chapter overview

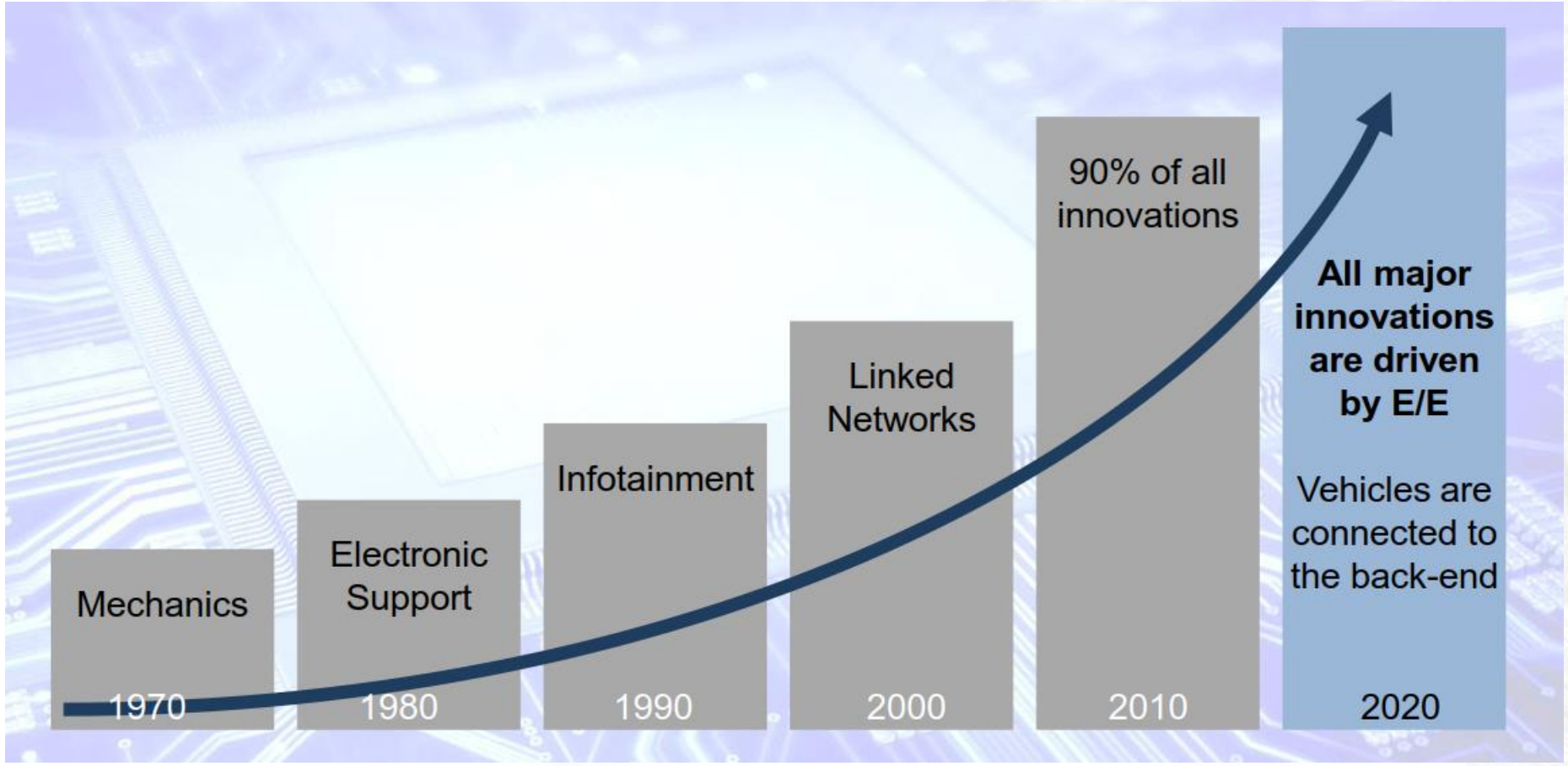


- Motivation for AUTOSAR
- Major AUTOSAR objectives
- AUTOSAR organization and evaluation of Standards
- AUTOSAR concepts
  - Software components (SW-C)
  - Virtual function bus (VFB)
  - Runtime Environment (RTE)
  - Basic Software (BSW)

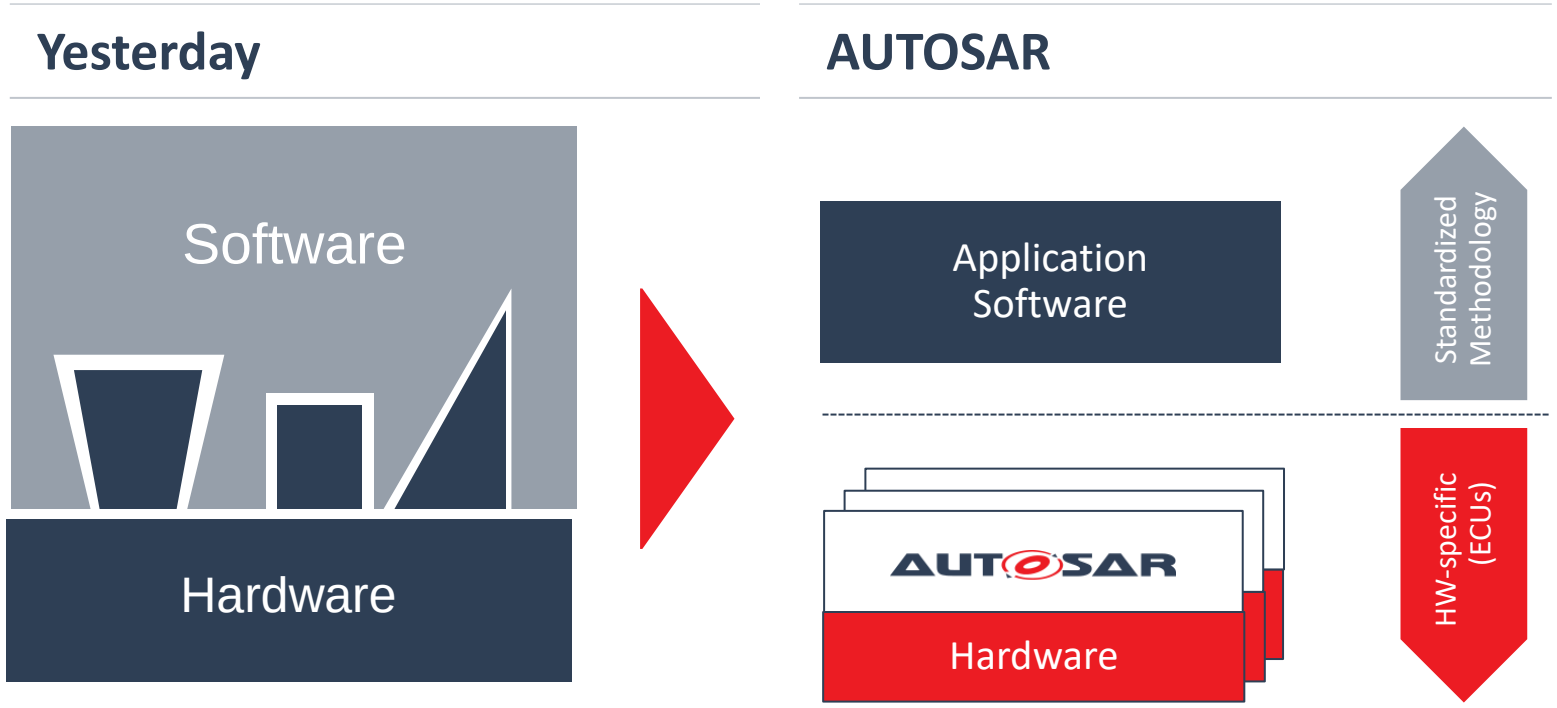
Motivation for AUTOSAR

Elektrobit

Technology Changes



Aims and benefits of using AUTOSAR



- Hardware and software – widely independent of each other
- Development can be de-coupled by horizontal layers, reducing development time and costs
- Reuse of software enhances quality and efficiency
- Companies cooperate in standardization and compete in implementation

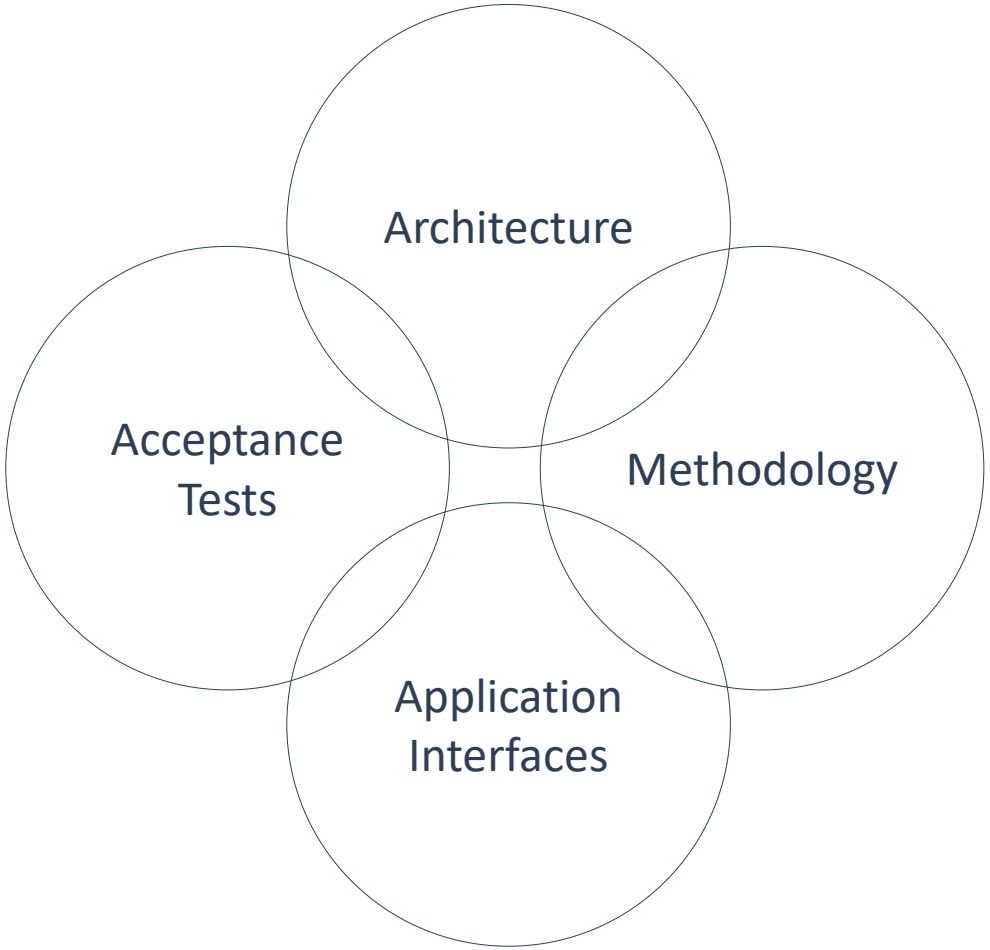
(AUTomotive Open System ARchitecture)

is a worldwide development partnership of vehicle manufacturers, suppliers, service providers and companies from the automotive electronics, semiconductor and software industry.

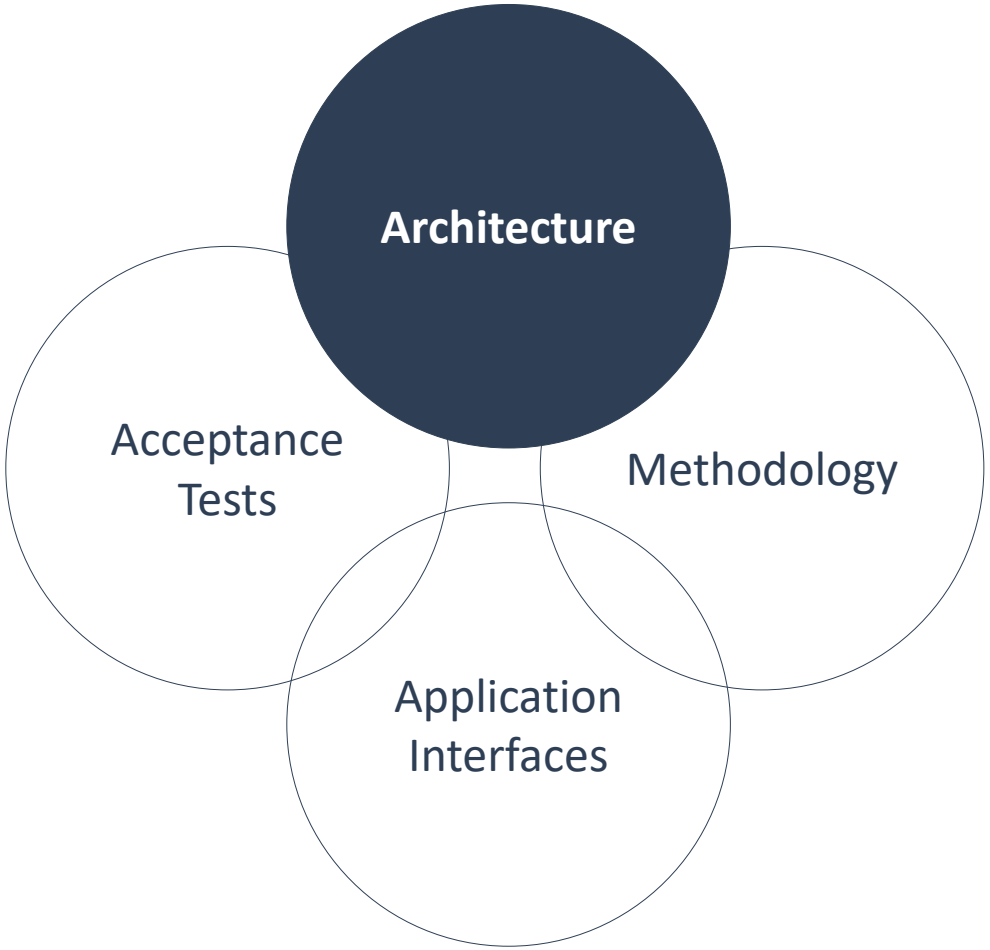
Major AUTOSAR objectives

Elektrobit

Main Working Topics

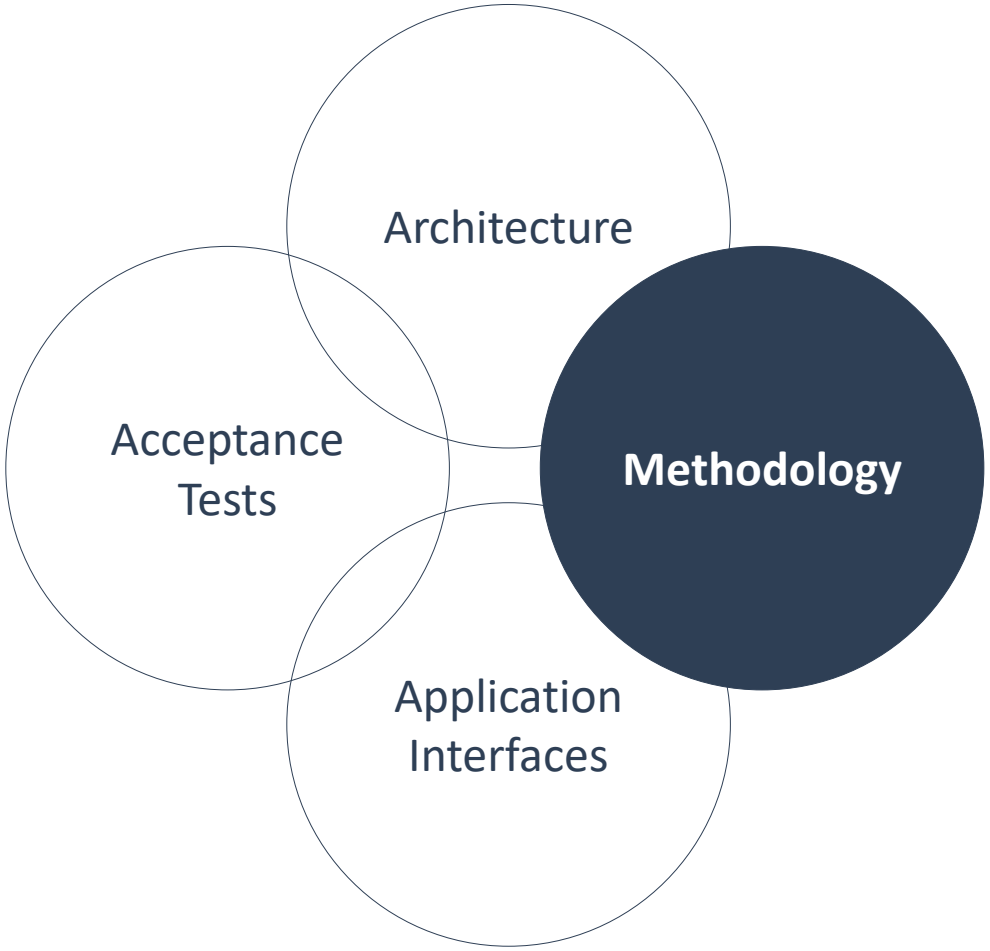


Main Working Topics



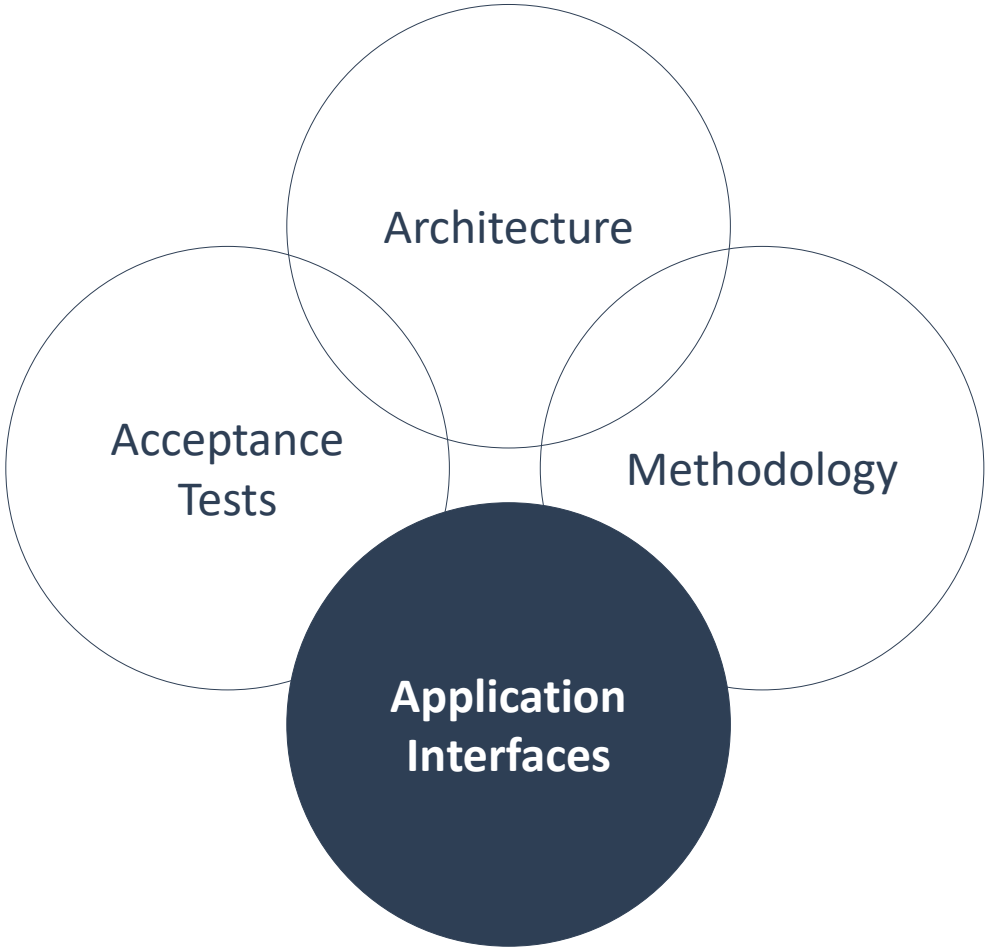
Software architectures including a complete basic software stack for ECUs – the so called AUTOSAR Basic Software – as an integration platform for hardware independent software applications.

Main Working Topics



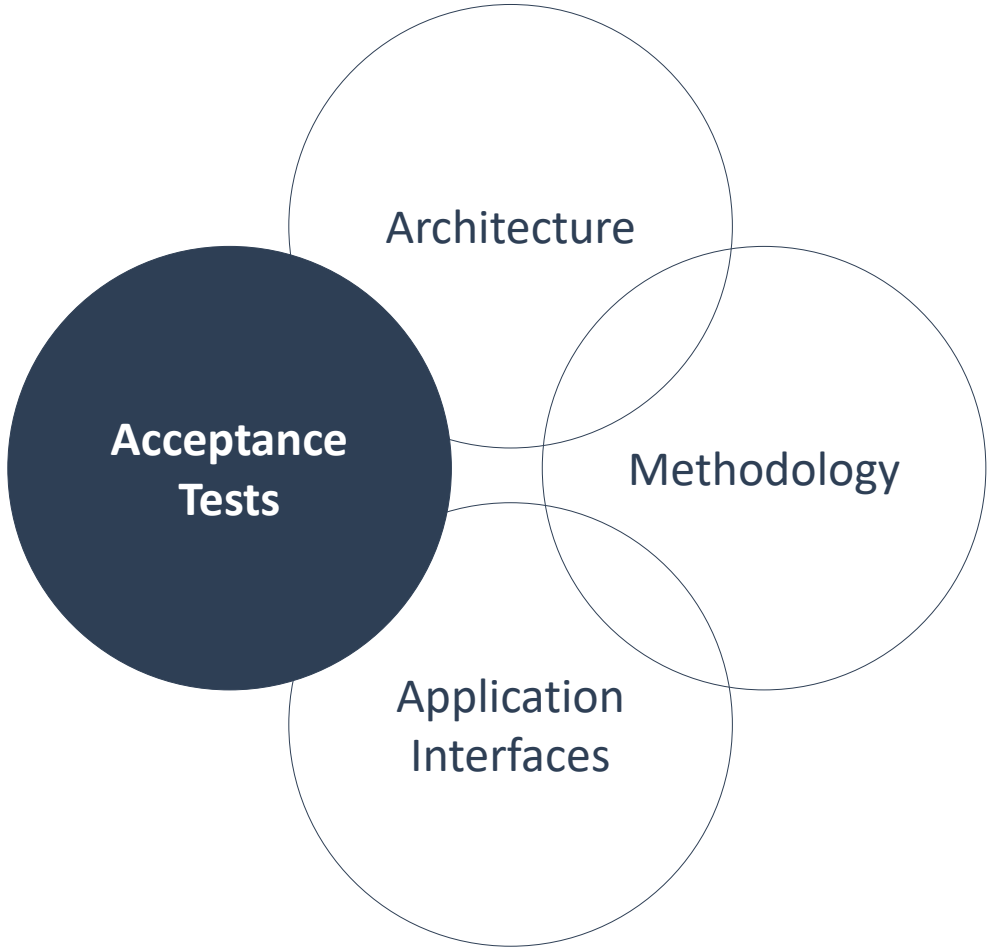
Defines exchange formats and description templates to enable a seamless configuration process of the basic software stack and the integration of application software in ECUs. It includes even the methodology how to use this framework.

Main Working Topics



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 Working Topics

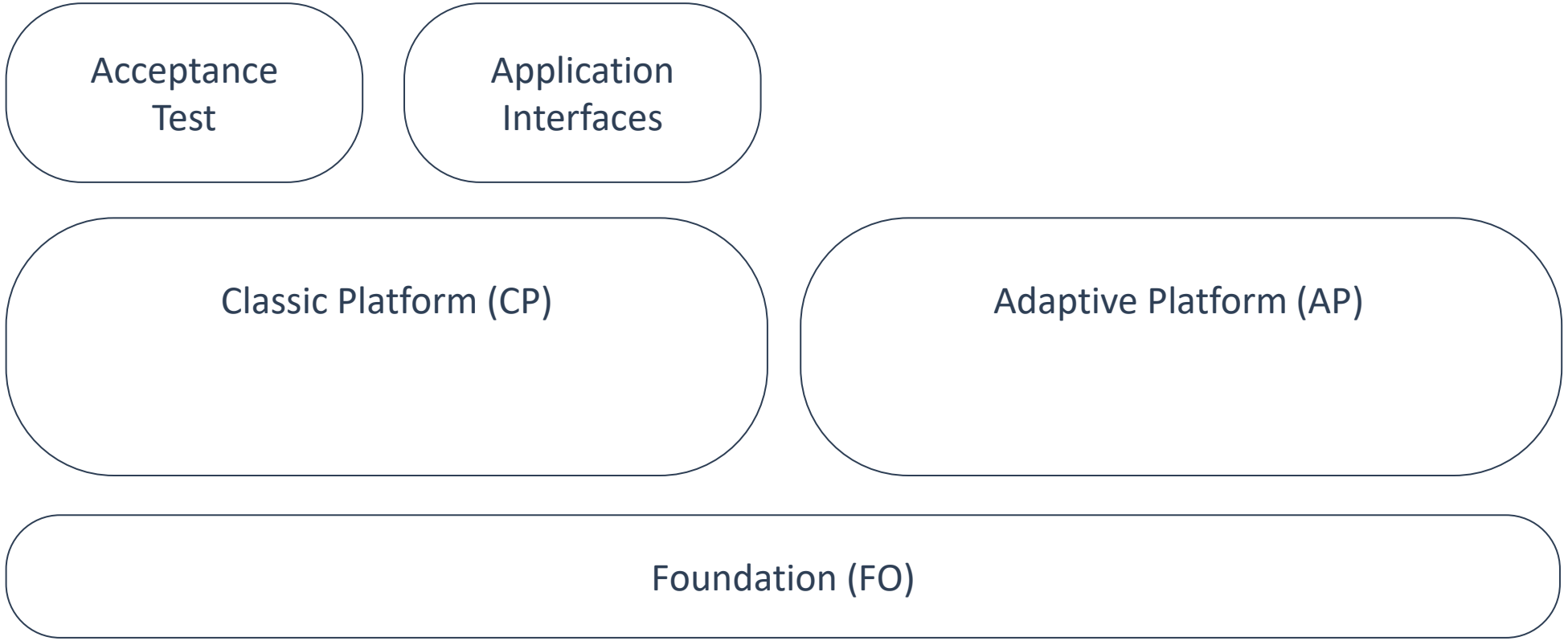


Specification of test cases intending to validate the behavior of an AUTOSAR implementation with AUTOSAR application software components or within one vehicle network.

AUTOSAR organization and evolution of Standards

Elektrobit

The platforms are organized by 5 AUTOSAR standards



AUTOSAR Classic Platform	AUTOSAR Adaptive Platform
Based on OSEK	Based on POSIX
Execution of code directly from ROM	App. is loaded from persistent memory into RAM
Same address space for all applications (MPU support for safety)	Each application has its own (virtual) address space (MMU support)
Optimized for signal-based communication (CAN, FlexRay)	Service-oriented communication
Fixed task configuration	Support of multiple (dynamic) scheduling strategies
Specification	Specification and code

AUTOSAR – Core Partners and Partners

9 Core Partners

BMW Group

BOSCH

Continental

Ford

DAIMLER

PSA GROUPE

GM

TOYOTA

VOLKSWAGEN AG

53 Premium Partners

General OEM

VOLVO TRUCKS · HONDA The Power of Dreams · TATA MOTORS · HYUNDAI · VOLVO · RENAULT · NISSAN · Great Wall

Generic Tier 1

Autoliv · assystem · HELLA · dSPACE · DENSO · JTEKT · LEAR CORPORATION · DELPHI · Panasonic · ZF · NEC · ThyssenKrupp · CNH INDUSTRIAL · Valeo

Standard Software

Mentor Graphics · EB · intel · MathWorks · KPIT · eSOL · VECTOR · SCSK · LUXOFT · NVIDIA

Tools and Services

ALTRAN · ETRI · itk ENGINEERING · L&T Technology Services · cea · Deloitte. · Green Hills SOFTWARE · TATA ELXSI LIMITED · LG Electronics · automotive engineering iAV · ETAS · DASSAULT SYSTEMES · sodius · ARC CORE · Fraunhofer

Semiconductors

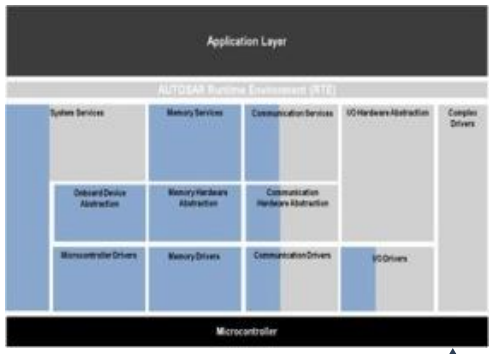
Infineon · RENESAS · life.augmented · NXP

38 Development Partners

ANSYS · Advanced Telematic SYSTEMS · PRQA Programming Research · avelabs · basicworx · E.S.R.LABS · INCHRON · rti · eJad · Timing Architects · PROVE & RUN · WAFER SPACE · BASELABS Data Fusion Results · itemis · silicon gears · lhp · easycore · OFFIS · REACH AUTOMOTIVE TECHNOLOGY · SPIRENT · GLIWA · FPT Fpt Software · C&S · OPENSYNERGY · Freetech · IP Camp · LAUTERBACH · IncQueryLabs · EVIDENCE · osb TTTech · SYMTA VISION · ixia · excelfore · nordsys · NCES · VALIDAS · seneos · AVIN SYSTEMS · PopcornSAR

127 Associate Partners
21 Attendees

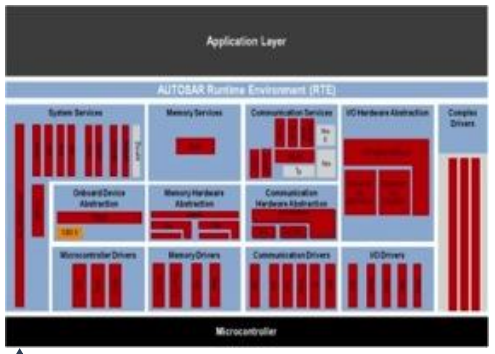
AUTOSAR Milestones



First release (1.0)

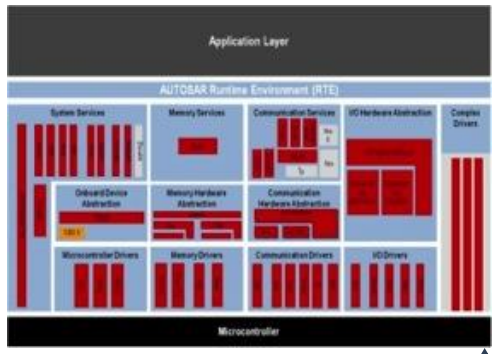
Release 2.0/2.1
Basic SW complete

Release 3.0/3.1
Additional features



Release 4.0
Feature enrichment

- Multicore support
- Functional safety
- Ethernet
- …



Release 4.1.1

- Timing analysis
- Diagnostics
- Acceptance testing
- …

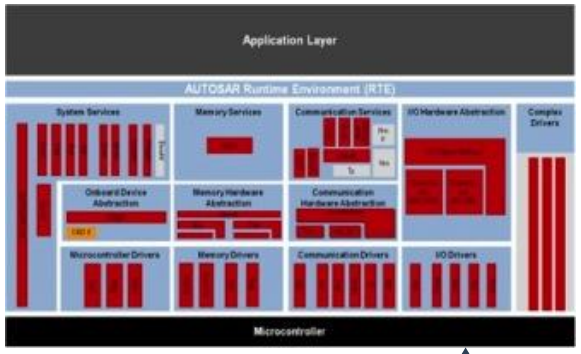
AUTOSAR founded

Derived applications

New development methods

2003 — 2004 — 2005 — 2006 — 2007 — 2008 — 2009 — 2010 — 2011 — 2012 — 2013

AUTOSAR Milestones



Release 4.2.1

- **Large data communication via Ethernet and CAN FD**
- …

Release 4.3.0

- Hardware Test Management on Startup and Shutdown
- **Crypto Interface**
- V2X Support
- Extended Buffer Access for Rapid Prototyping
- SOME/IP Transport Protocol Decentralized Configuration

Release 4.3.1

- Quality
- Interface Module for Ethernet and IP testing
- Macro Encapsulation Of Library Calls
- Error Detection and Correction for Communication

Release 4.4

- Remote Event Communication Manager
- Harmonization of Physcial Units for ASW and BSW Based on ASAM
- Security Policy Manager Module
- AUTOSAR Real Time Interface
- RTE Implementation Plug-Ins
- LIN-Support for LIN slave
- Ethernet Wake on data line
- Formal Model Query and Blueprint Derivation Mechanisms
- Bus-Mirroring
- Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding

2013 — 2014 — 2015 — 2016 — 2017 — 2018

AUTOSAR Milestones

Release 19-11

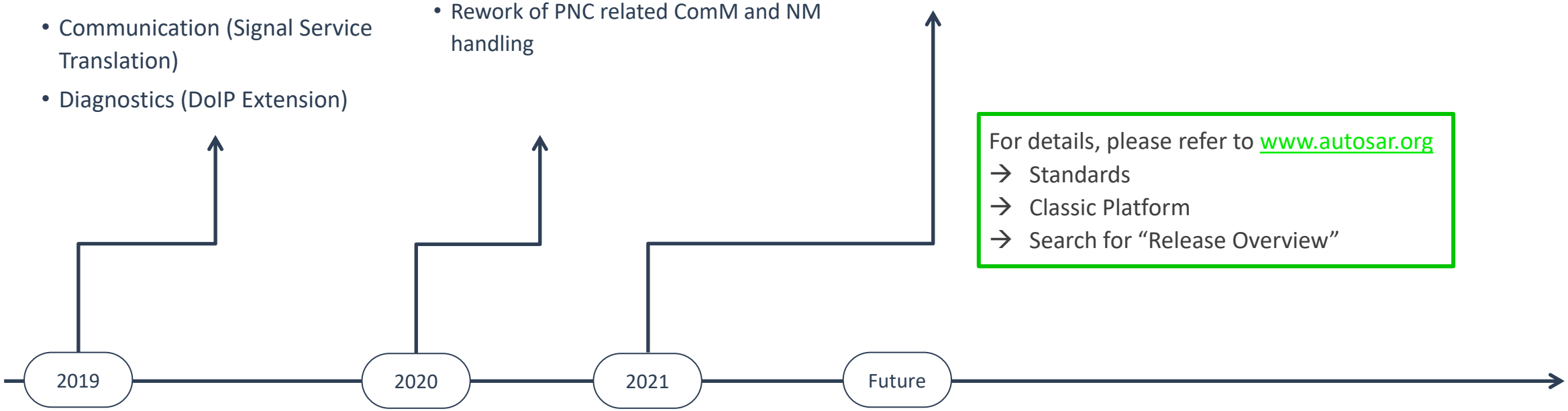
- BSW Multicore Distribution
- Non-Volatile Data Handling Enhancements
- Firmware over the Air
- IPSec Protocol
- Communication (Signal Service Translation)
- Diagnostics (DoIP Extension)

Release 20-11

- Ethernet: Wakeup On Dataline, 10BASE-T1S
- Security: Intrusion Detection System Manager (IdsM)
- Classic Platform Flexibility
- Rework of PNC related ComM and NM handling

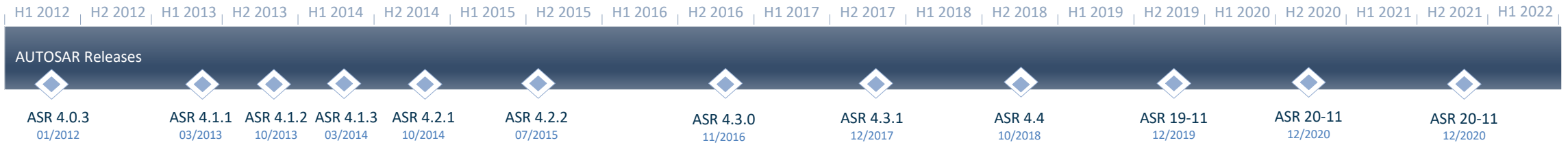
Release 21-11

- Functional Safety: e.g. Mode Dependent Configuration, System Health Monitoring
- Data Distribution Service (DDS) Security
- Memory Stack Rework



For details, please refer to www.autosar.org
→ Standards
→ Classic Platform
→ Search for “Release Overview”

AUTOSAR releases / EB tresos ACG Versions



ACG 7

ACG7 supports selected RfCs and selected Features based on AUTOSAR 4.0.3, 4.1.x and 4.2.x

ACG 8

ACG8 supports selected RfCs and selected Features based on AUTOSAR 4.0.3, 4.1.x , 4.2.x, 4.3.x, 4.4.x, 19-11, 20-11

Current Release

Implementation of the AUTOSAR Classic Platform → Autocore Generic (ACG)

8.7.0 8.7.1 8.7.2 8.8.0 8.8.2 **8.8.4**

Implementation of the corresponding tooling → EB tresos Studio

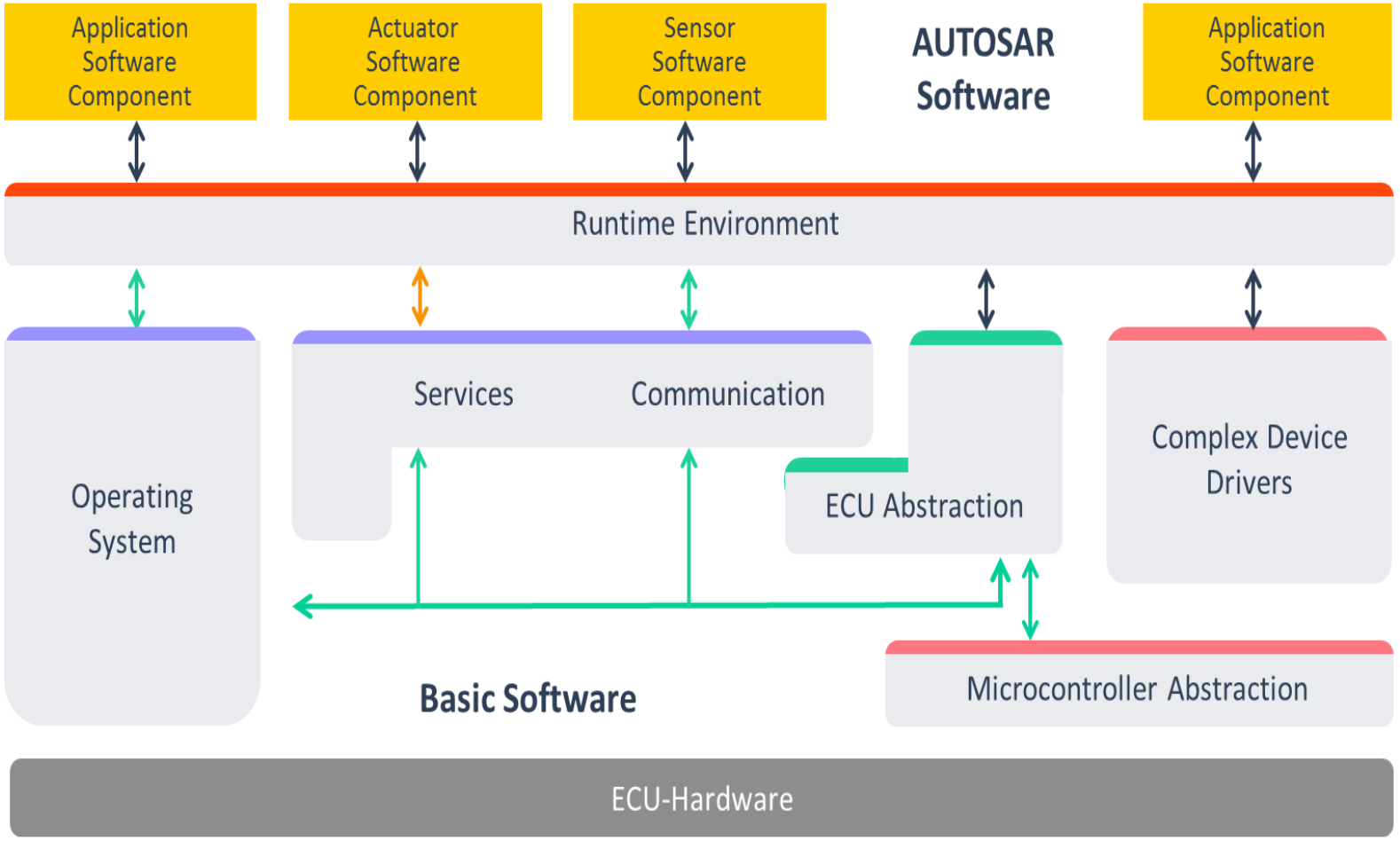
26.1 26.2 27 27.1 28.0 **28.2**

AUTOSAR Concepts

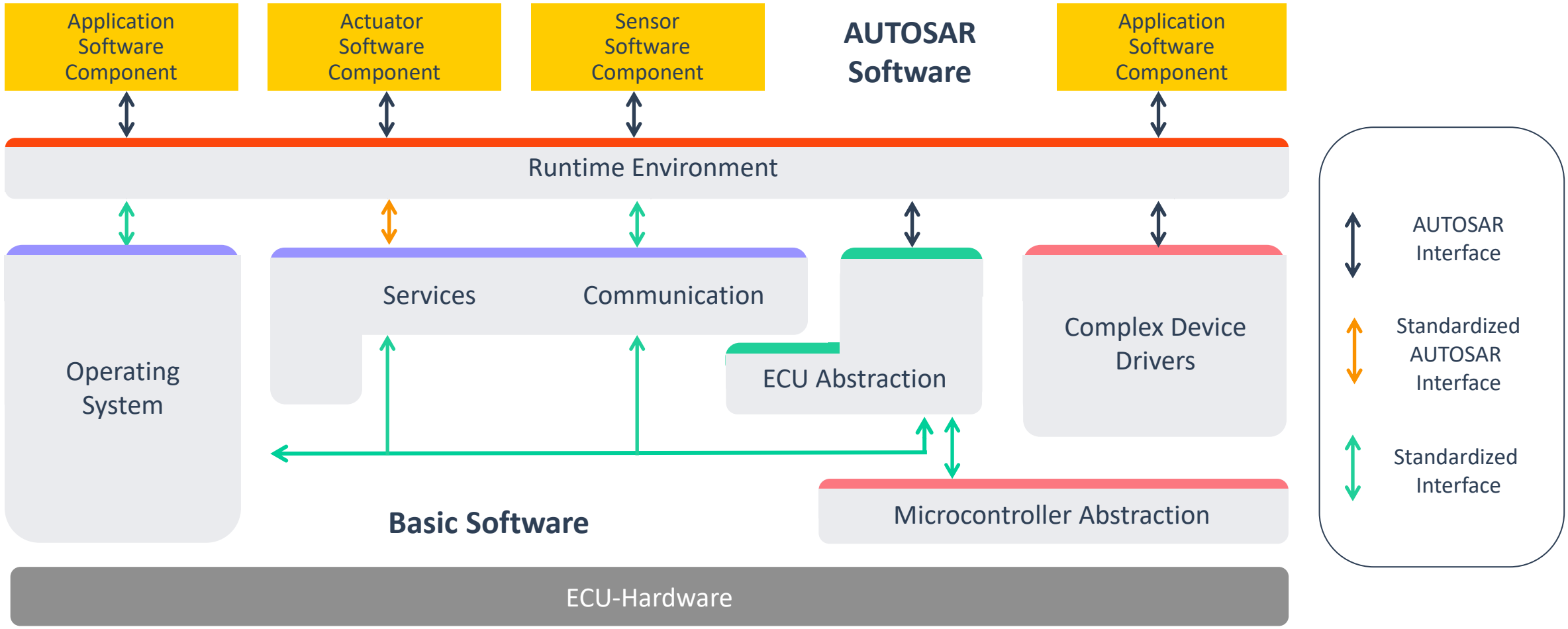
Elektrobit

Software architecture of AUTOSAR Classic Platform

- An AUTOSAR application consists of one or more (interconnected) **Software components (SW-C)**
- **Run Time Environment (RTE)** is a communication centre for inter- and intra-ECU information exchange.
  It realizes the communication defined in the **Virtual Functional Bus (VFB)** on one specific ECU
- The **Basic Software (BSW)** is a standardized software layer that provides standard ECU functionality (OS, low level drivers, bus-communication, diagnostics, memory management etc.)

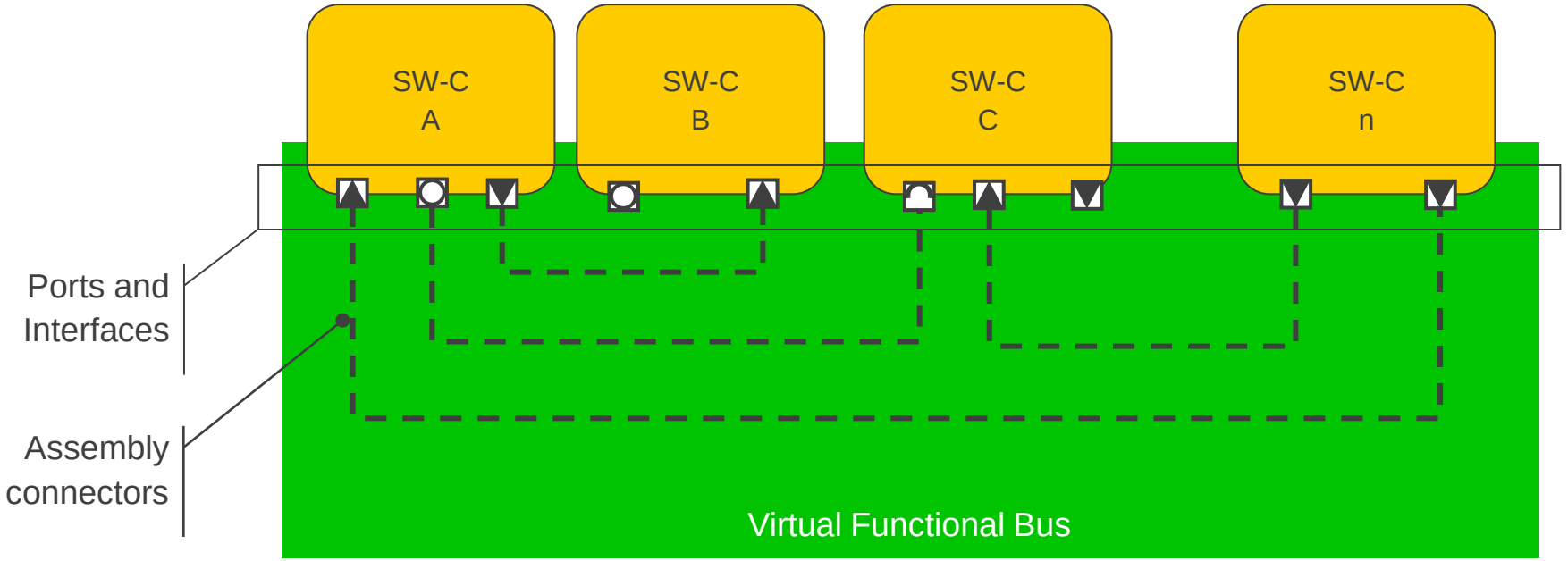


Software architecture of AUTOSAR Classic Platform



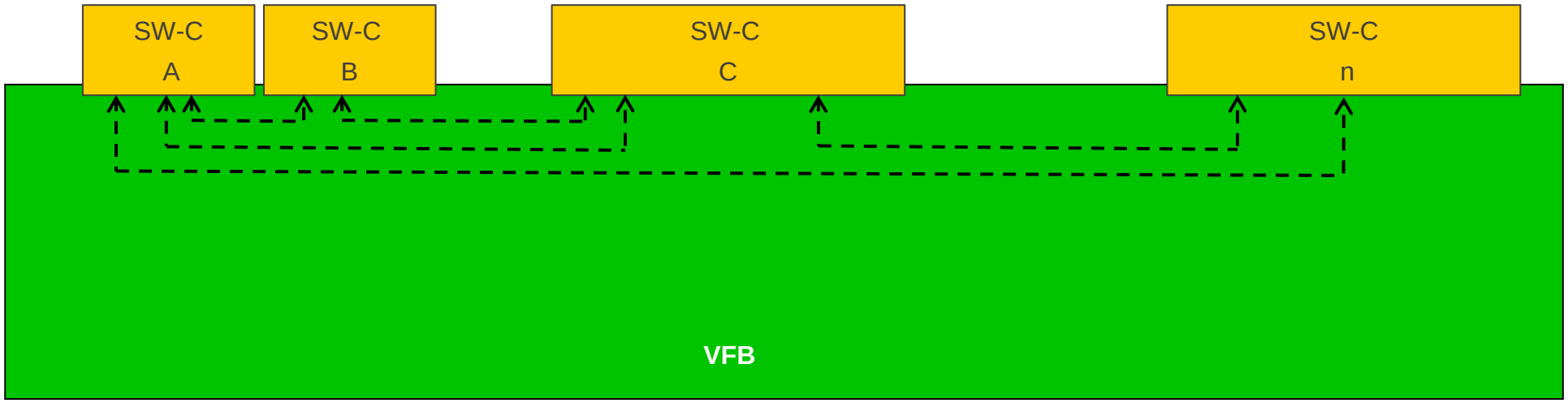
Software Components and Virtual Function Bus

- Software components (SW-C): Application software to run in an AUTOSAR system
- Communication between SW-Cs through well defined ***Ports*** and ***Interfaces*** using ***Assembly connectors***
- The High-level communication abstraction is called **Virtual Function Bus** (VFB)



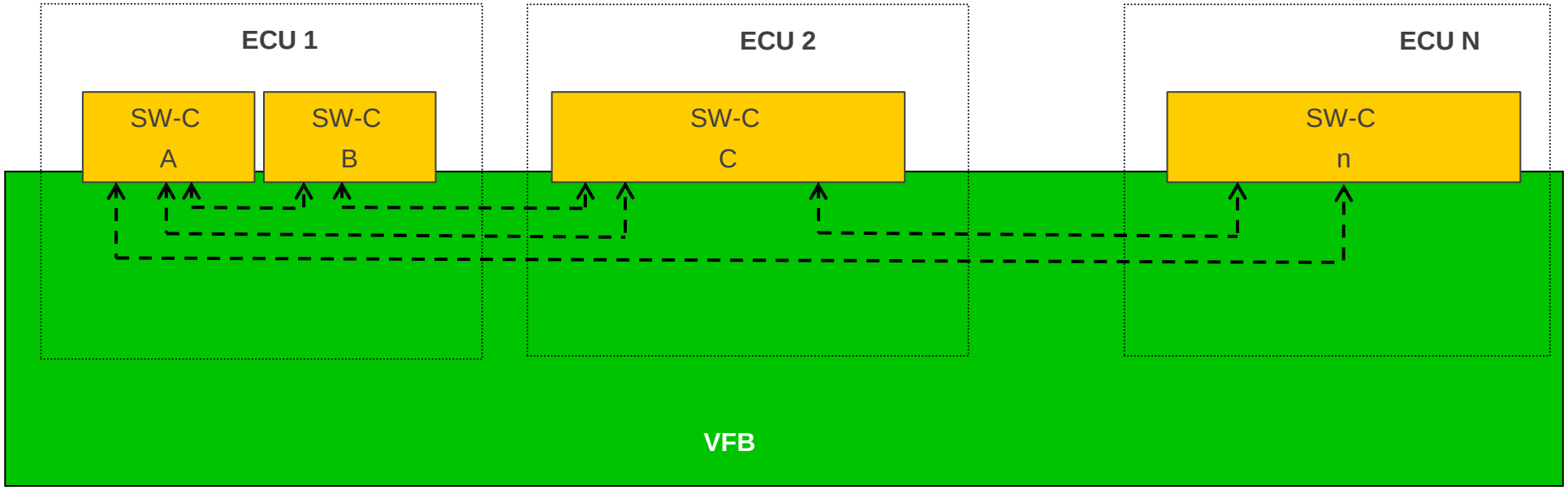
Virtual Function Bus

- Simplified picture of the VFB (without Ports)



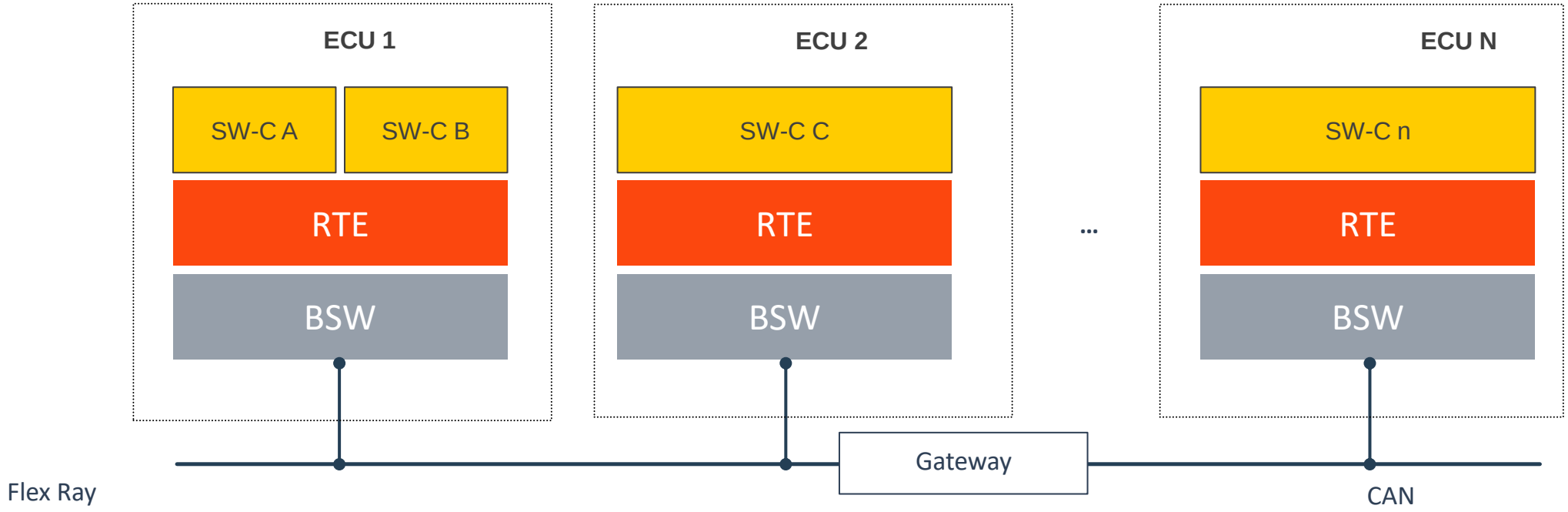
Virtual Function Bus

- During system design phase, the SW-Cs are partitioned to ECUs (SW-C to ECU mapping)
- This results in two different types of communication paths:
  - ***Intra-ECU*** (inside one ECU)
  - ***Inter-ECU*** (between different ECUs)



Runtime Environment (RTE)

- The ***Runtime Environment*** (RTE) is the only interface to the SW-Cs
- The RTE implements the VFB on each ECU
- The RTE uses Network (CAN/Ethernet/FlexRay/LIN) buses for inter-ECU communication via the ***Basic Software Layer*** (BSW)



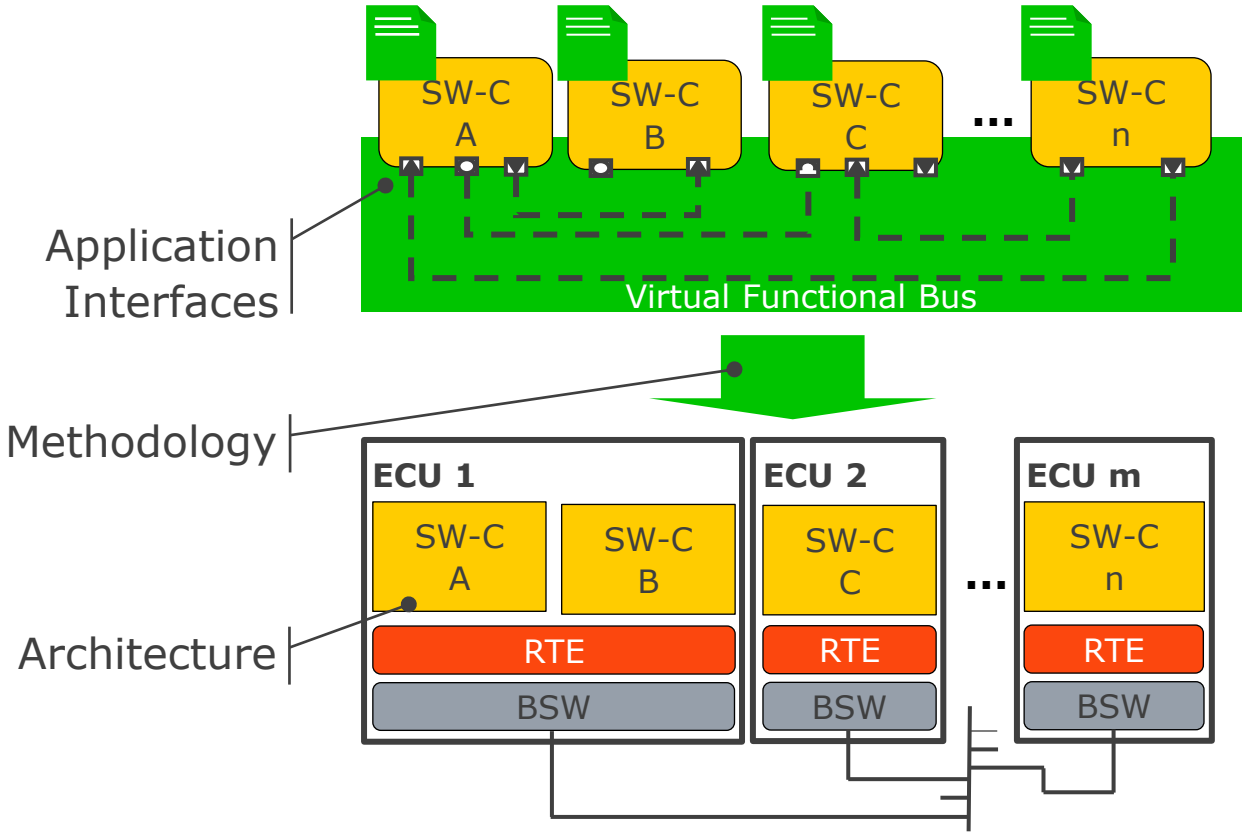
AUTOSAR Concepts - Summarized

Elektrobit

Software Components and Virtual Function Bus

AUTOSAR defines four key concepts:

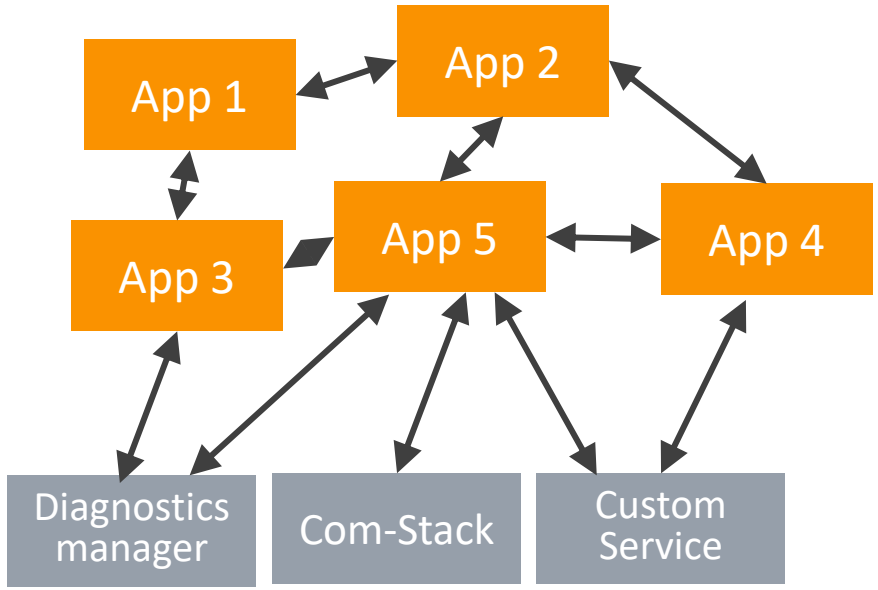
- ***Software components (SW-C)***
  – A piece of software to be run in an AUTOSAR system
- ***Virtual Functional Bus (VFB)***
  – High level communication abstraction
- ***Run Time Environment (RTE)***
  – Implements the VFB on one ECU
- ***Basic Software (BSW)***
  – Standard software for standard ECU functionality (OS, communication, memory, hardware drivers, diagnostics etc)



Virtual Function Bus and Runtime Environment (RTE)

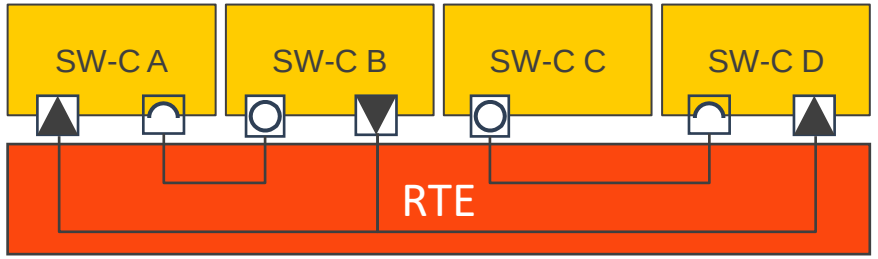
Non-AUTOSAR

- Hard-coded dependencies between Applications and services



AUTOSAR

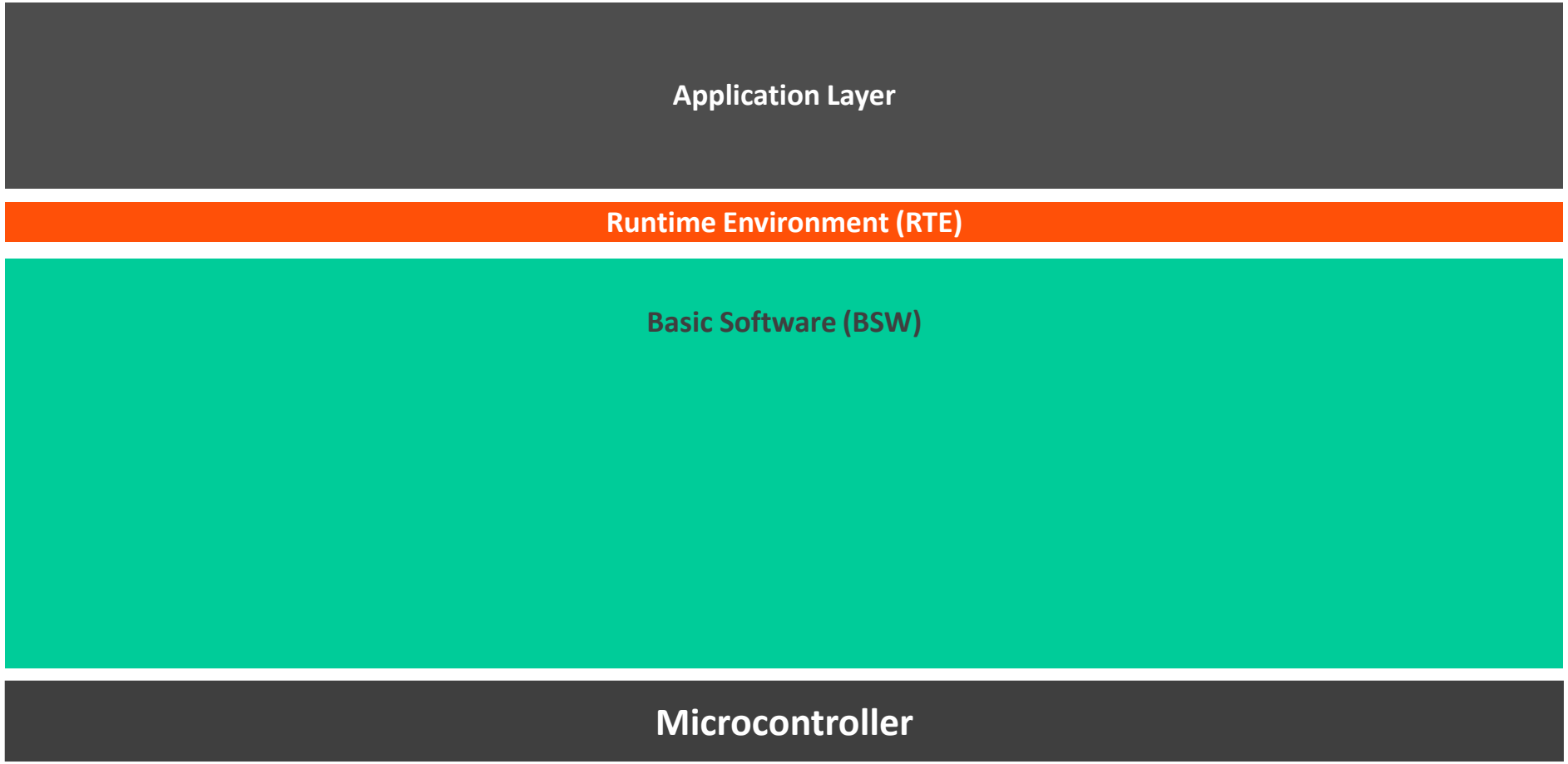
- Application communications exclusively with the RTE using well defined APIs
- No difference between internal communication and bus communication → ***Relocatability and reuse!***



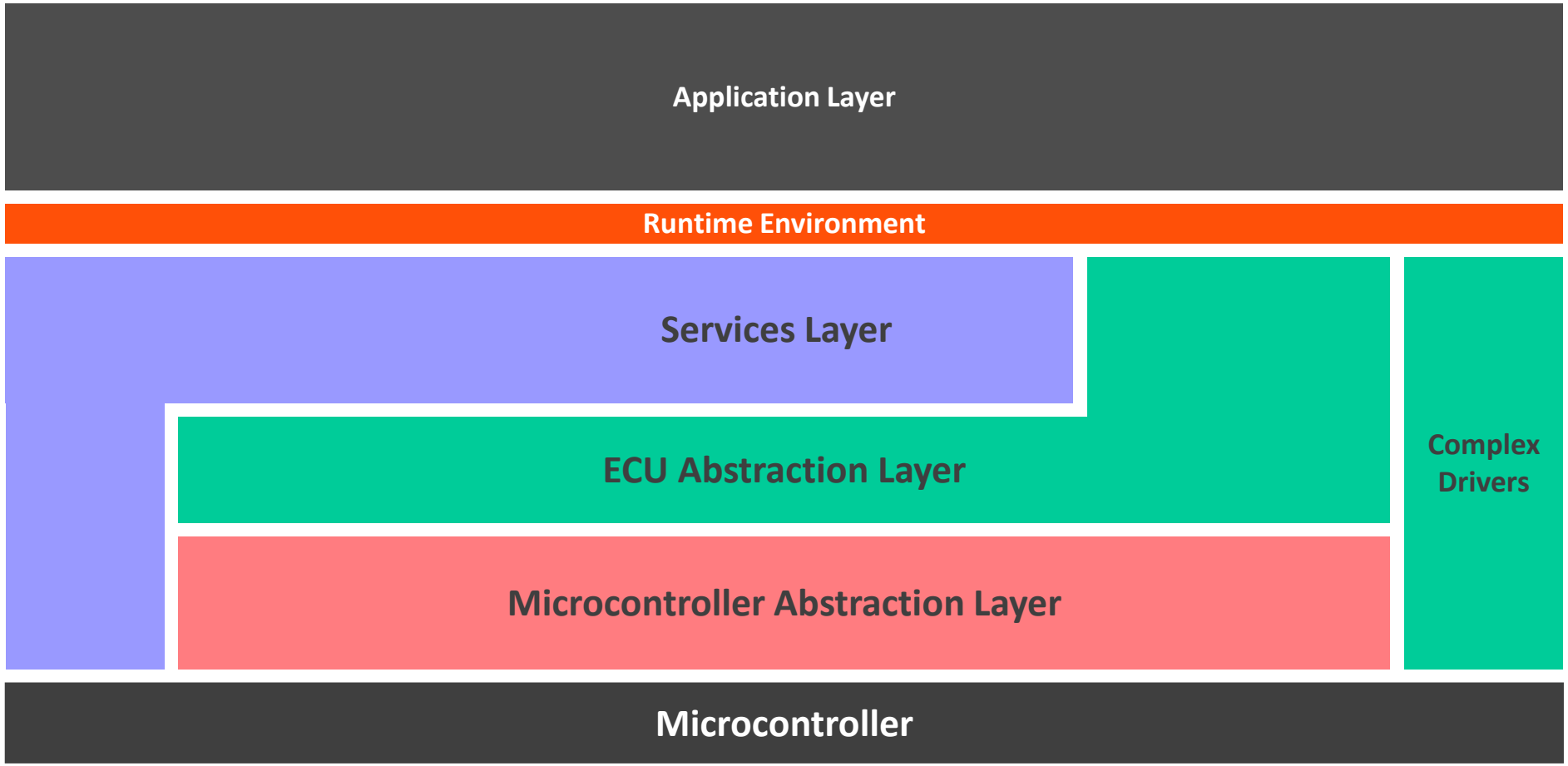
Basic Software (BSW)

Elektrobit

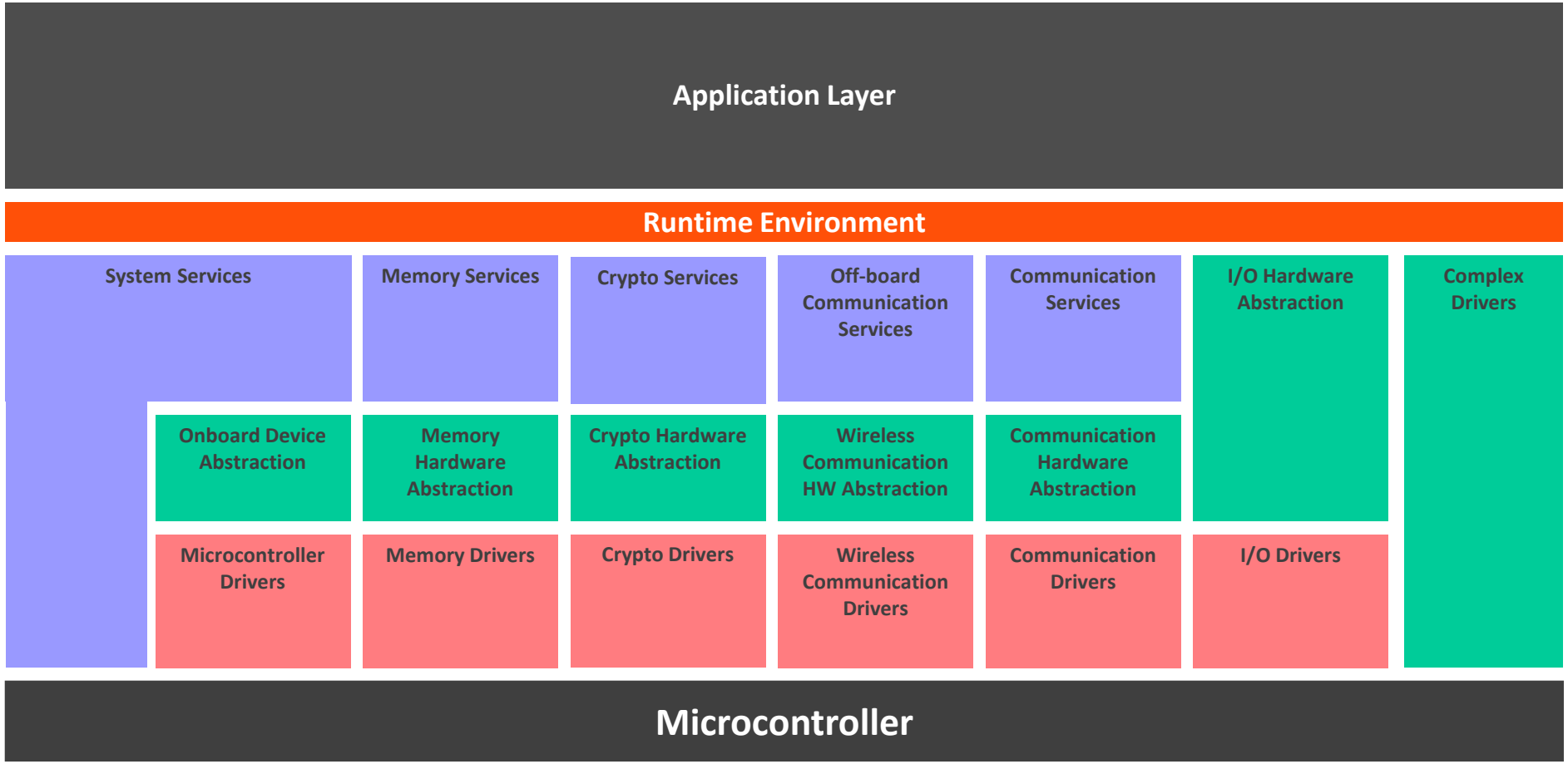
Overview of Software Layers (Top view)



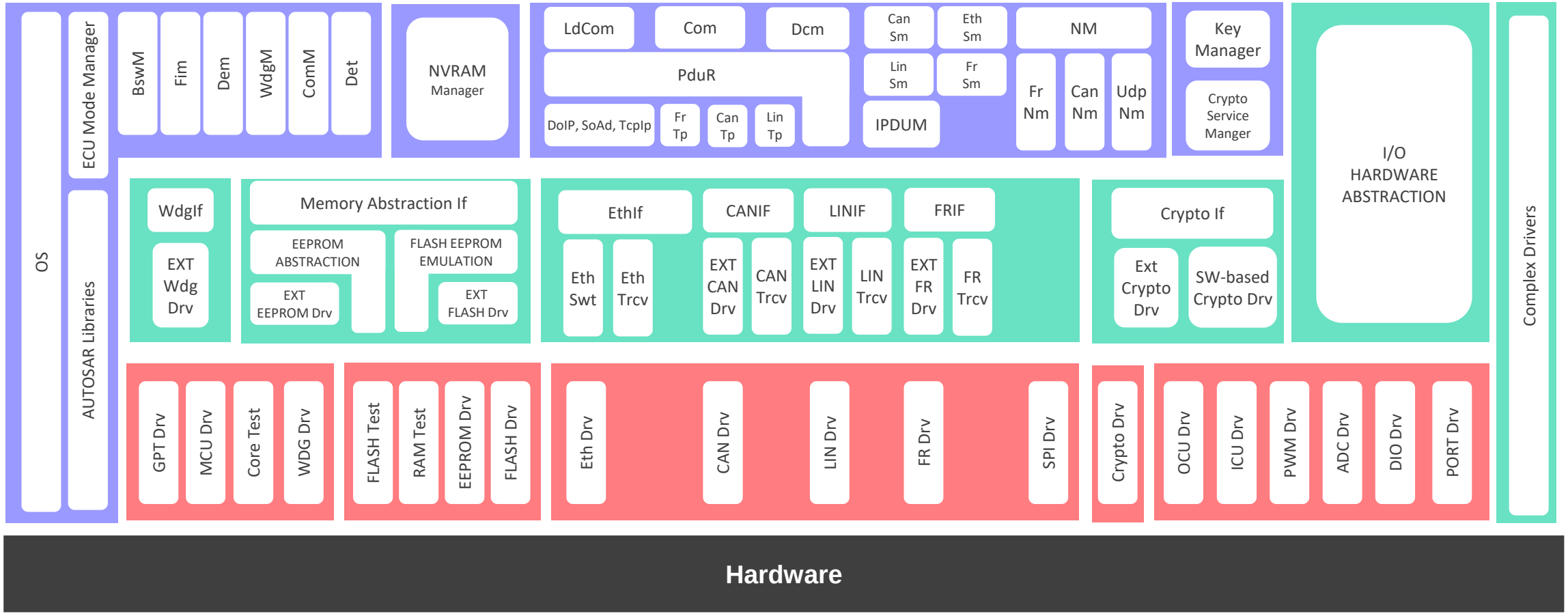
Basic Software – Layers



Basic Software – Functional groups



Basic Software – Detailed view



Summary

- Motivation for AUTOSAR
- Major AUTOSAR objectives
- AUTOSAR organization and evaluation of Standards
- AUTOSAR concepts
  - Software components (SW-C)
  - Virtual function bus (VFB)
  - Runtime Environment (RTE)
  - Basic Software (BSW)

Get in touch!

Elektrobit

sales@elektrobit.com
www.elektrobit.com