AUTOSAR – AUTOMOTIVE TECHNOLOGY IN PRACTICE

JAROSLAV KADLEC

2KSYS

2KSYS

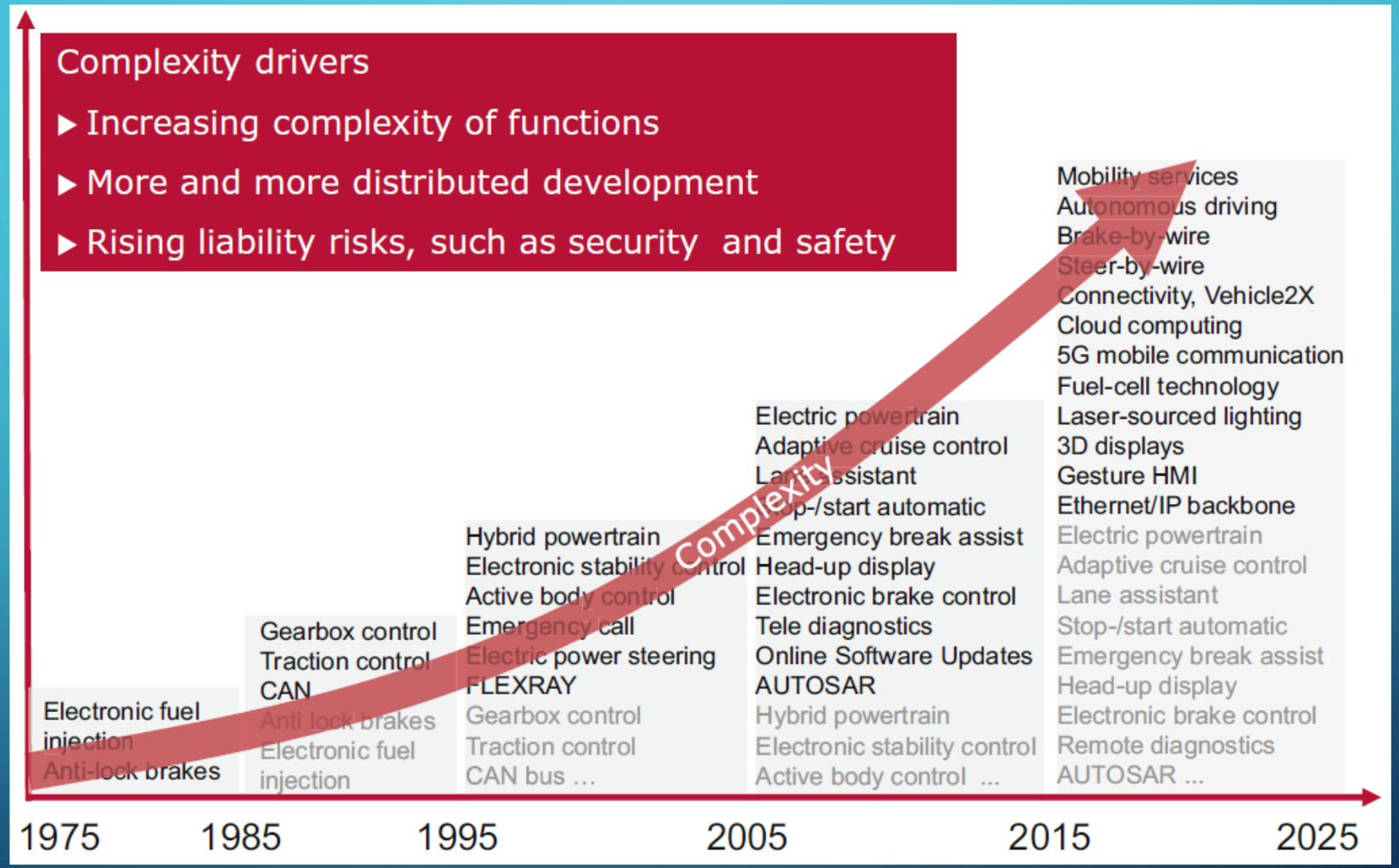
- History
  - Founded in 2015 as a start-up company
  - First projects oriented to custom HW design and SW development for microcontrollers
  - 2017 designed Control electronic for Combustion Engine in JetSurf
  - 2018 moved to TITC - Technology Innovation Transfer Chamber
  - 2018 first SW within AUTOSAR standard

- Main customers:

JETSURF ima MSR engines

2KSYS s.r.o.

INTRODUCTION - HISTORY OF AUTOMOTIVE



Source: Automotive Software Architectures: An Introduction 1st ed. 2017 Edition by Miroslaw Staron

2KSYS s.r.o.

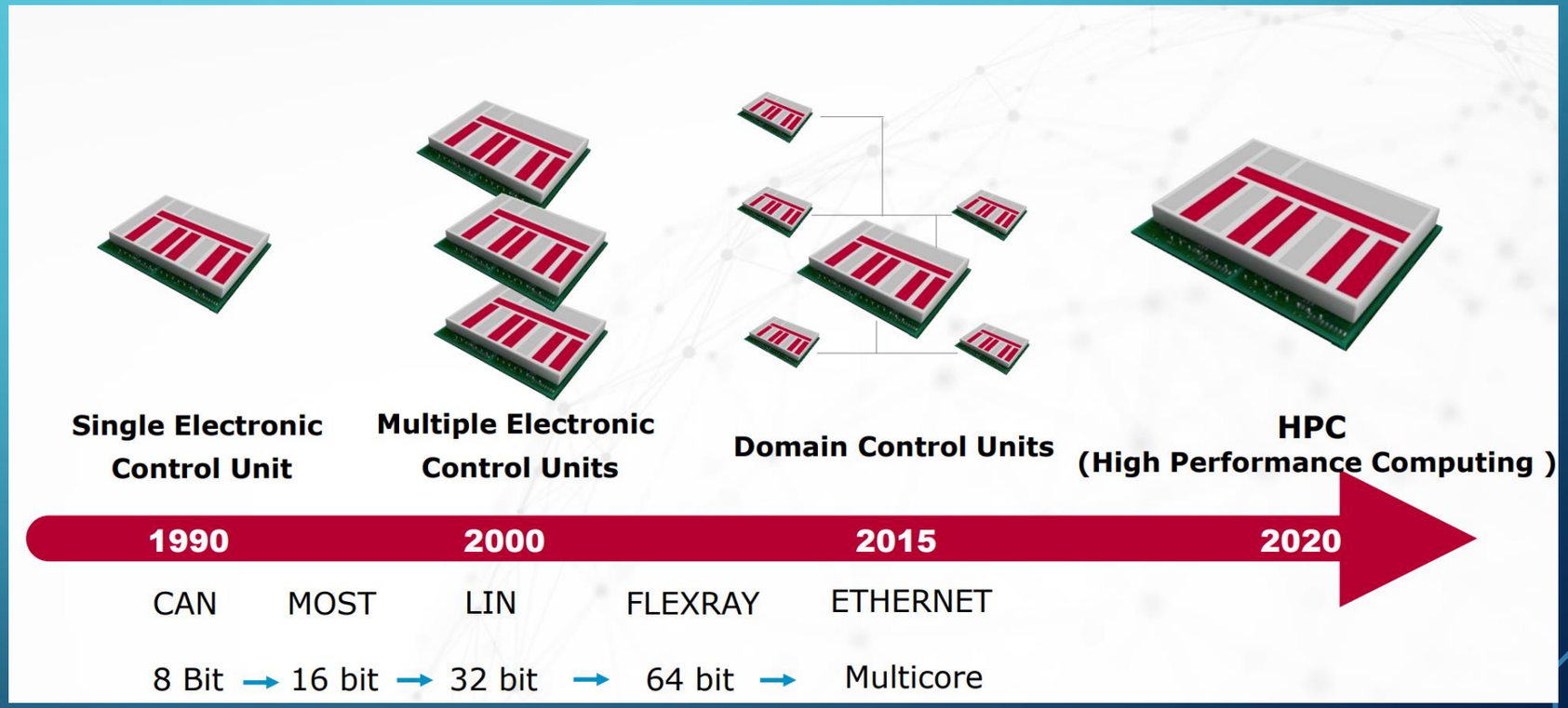
INTRODUCTION - HISTORY OF AUTOMOTIVE

- 1970 – 1 ECU, 1 software
- 1990 - Distributed software
- 2002 - Advanced Driver Support, Adaptive Tempomat, Emergency breaking, introduced AUTOSAR
- 2007 - SW size measured in megabytes
- 2010 – C2X communication, autonomous driving, 10 million lines of code*
- 2016 - SW size measured in gigabytes
- today
  - 150 million lines of code*
  - Modern vehicles have more than 50 ECUs
    - Premium cars having more than 100 ECUs
  - Some functions, such as engine control or dynamics, are hard real-time functions, with reaction times going down to a few milliseconds
  - Practically all other functions, such as infotainment, demand at least soft real-time behaviors.

*McKinsey&Company Report Feb'18 'Rethinking car software and electronic architecture'

FUTURE TRENDS IN AUTOMOTIVE

- e-Mobility
- Connectivity and cooperation, IoT, Weareables
- Autonomous functions
- Shared cars
- Big data
- C2X
- Cloud technologies
- Security
- Safety



*Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

2KSYS s.r.o.

TRENDS IN AUTOMOTIVE SOFTWARE DEVELOPMENT

- Heterogenity of software
  - from highly safety-critical
  - up to infotainment
- Distribution of development - OEMs and suppliers
- Distribution of software - number of ECUs, need of coordinate SW distribution, OTA SW updates
- Variants and configurations - different requirements on the same car in different countries
- Unit-based cost models - competitive market pushes unit prices of all components down
- Other trends
  - speed of SW development
  - data-driven development
  - ecosystems

ORGANIZATION OF AUTOMOTIVE SOFTWARE SYSTEMS

- Each automotive area has its own requirements for computational speed, reliability, security, safety, flexibility, and extensibility
- Past
  - Each car manufacturer developed its own way of organizing software systems
    - Common signs
      - Similar organization of the electrical and SW systems
      - V development model
- Now
  - High effort in creation Common and standardized architecture
    - This resulted in AUTOSAR

AUTOSAR

- AUTOSAR - **AUTO**motive **O**pen **S**ystem **AR**chitecture
- Improve software quality and reduce costs by RE-USE
  - Re–use of functions across carlines and across OEM boundaries
  - Re–use of development methods and tools
  - Re–use of basic software

Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

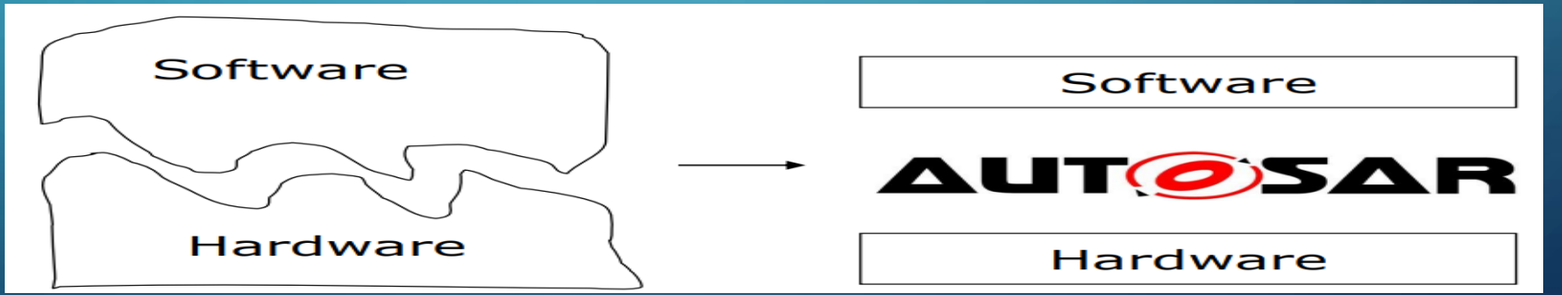
HARDWARE / SOFTWARE PLATFORMS

PAST

- Many different hardware platforms are used
  - Embedded systems traditionally do not support full hardware abstraction
  - Limited modularity of the software
- Poor reusability: Software must often be rewritten from scratch when hardware (processor type) is changed
- Variability: Suppliers have to support a large variety of OEMs and vehicle platform variants with their software

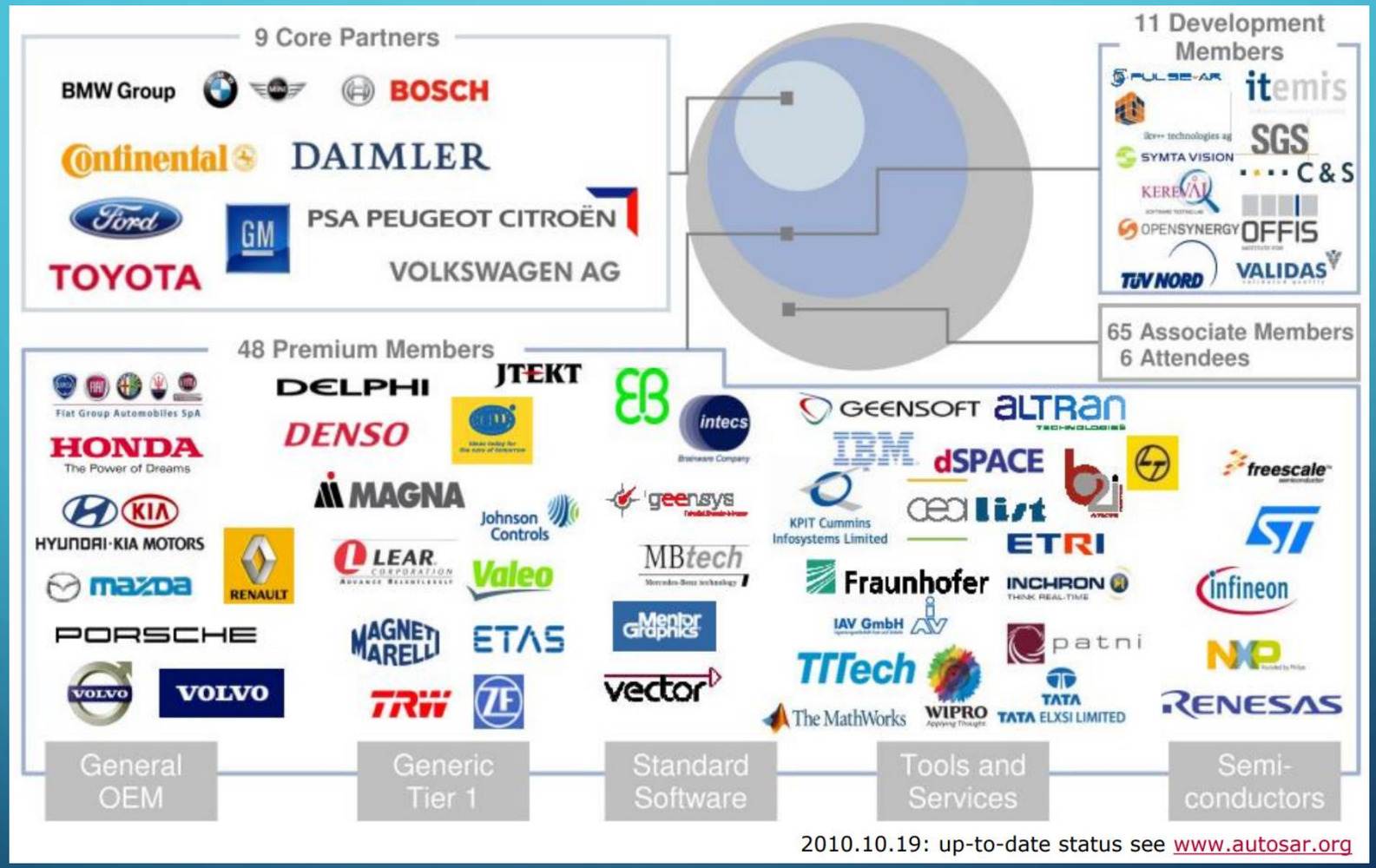
NOW

- **Serviceability** over the entire product life cycle
  - Software updates and upgrades over the entire life of a vehicle
- **Abstraction** of hardware from software, making development more flexible.
- Shift development activities from implementation to **configuration**
- Improvement in software **quality** by standardized BSW
- **Competition** is focused onto OEM-relevant features
- **Reusability** of functions across vehicle networks and across OEM boundaries.



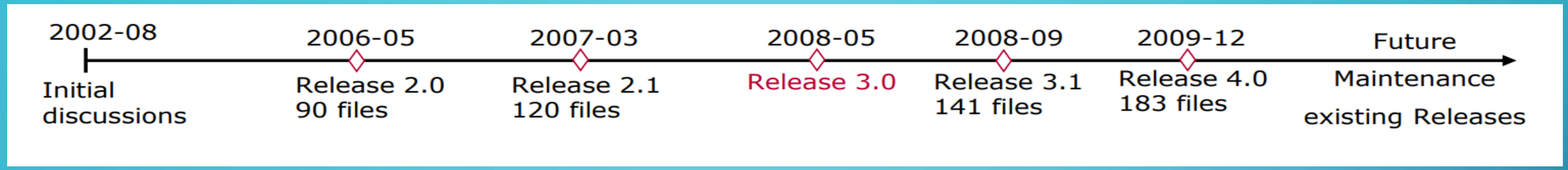
Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

AUTOSAR PARTNERSHIP



Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

HISTORY



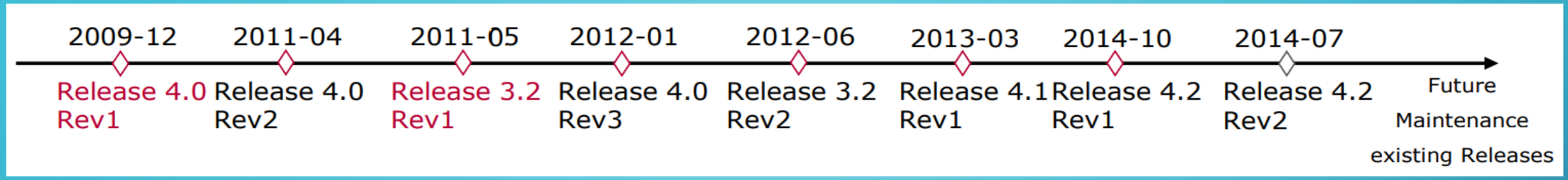
- AUTOSAR 2.0
  - specifications on components (BSW and RTE)
- AUTOSAR 2.1
  - includes specifications on development methodology and templates
  - first descriptions for application interfaces (body and interior electronics)

- AUTOSAR 3.0
  - > 20 compositions from the body, powertrain, and chassis domain
  - Specifications of standardized application interfaces (powertrain, chassis domain)
  - harmonized the concept for wakeup of ECUs and startup of networks
  - state managers for the bus systems CAN, LIN, and FlexRay added

- AUTOSAR 3.1
  - OBDII (new revisions of the DCM, DEM, FIM specifications,…)

Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

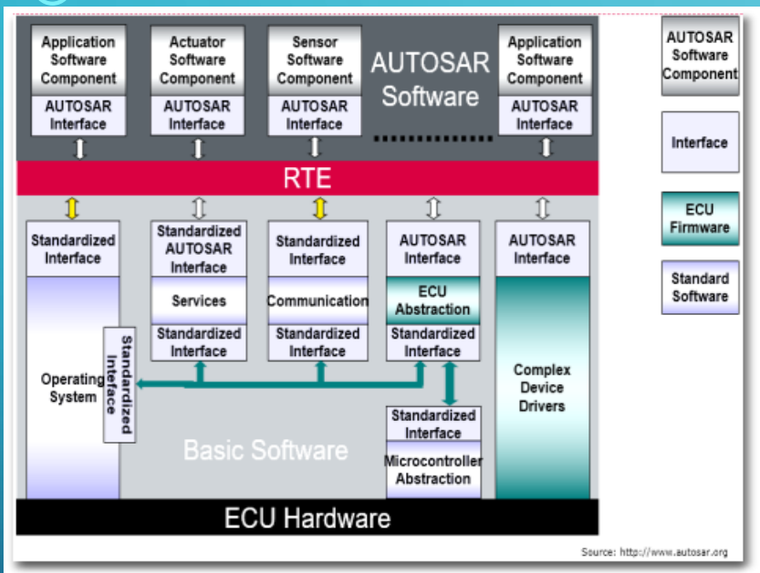
HISTORY



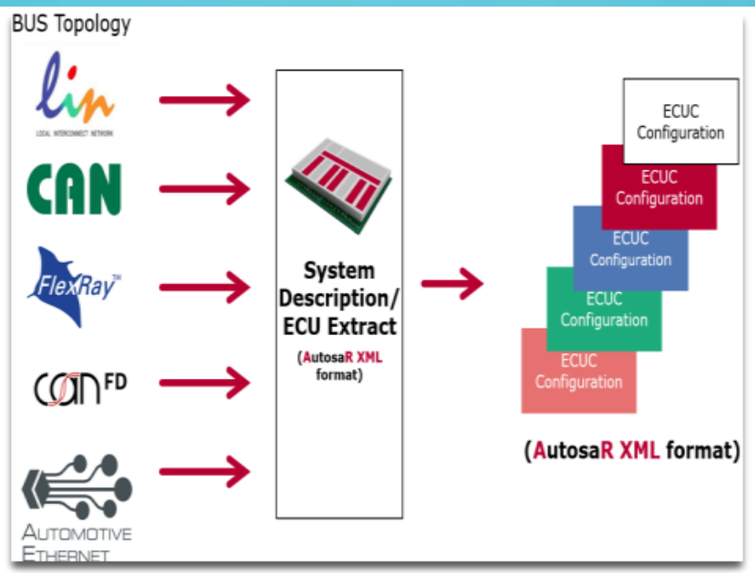
- AUTOSAR 4.0 Rev1 (2009-12)
  - Functional Safety (Memory Partitioning, Time Determinism, Program Flow Monitoring, E2E, BSWM Defensive Behavior, Dual µC, E-Gas Monitoring)
  - Architectural Improvements (Error Handling, Multi Core, Boot loader Interaction)
  - RTE enhancement (Triggered Events, Integrity and Scaling at Ports, API Enhancement)
  - COM (LIN 2.1 Spec, FR 3.0 Spec , FlexRay ISO TP, XCP for ASR, Large Data Types, TCP/IP + DoIp, J1939Tp, TTCAN)
  - Functional (NM coordination, ASR Scheduler)
  - Conformance Test specifications (CT Specs, only Rev1 and Rev2, but not >Rev2)
- AUTOSAR 3.2 Rev1 (2011-05)
  - Partial Networking
  - Robustness Features (state manager modules)
  - Improvement of error handling (e.g. production vs. development errors)
  - Back-porting of AUTOSAR 4.0 features into AUTOSAR 3.2
    - Parts of the Safety Concept (E2E communication protection)
    - Extended CDD Concept
    - BSW Mode Manager
    - FlexRay ISO TP

Source: VECTOR - Why AUTOSAR V0.1 | 2019-07-09

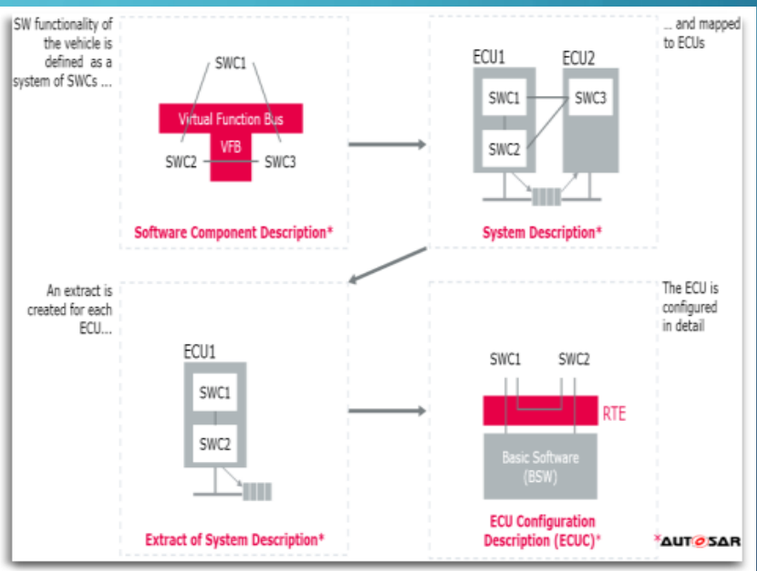
STANDARDIZATION



Interfaces



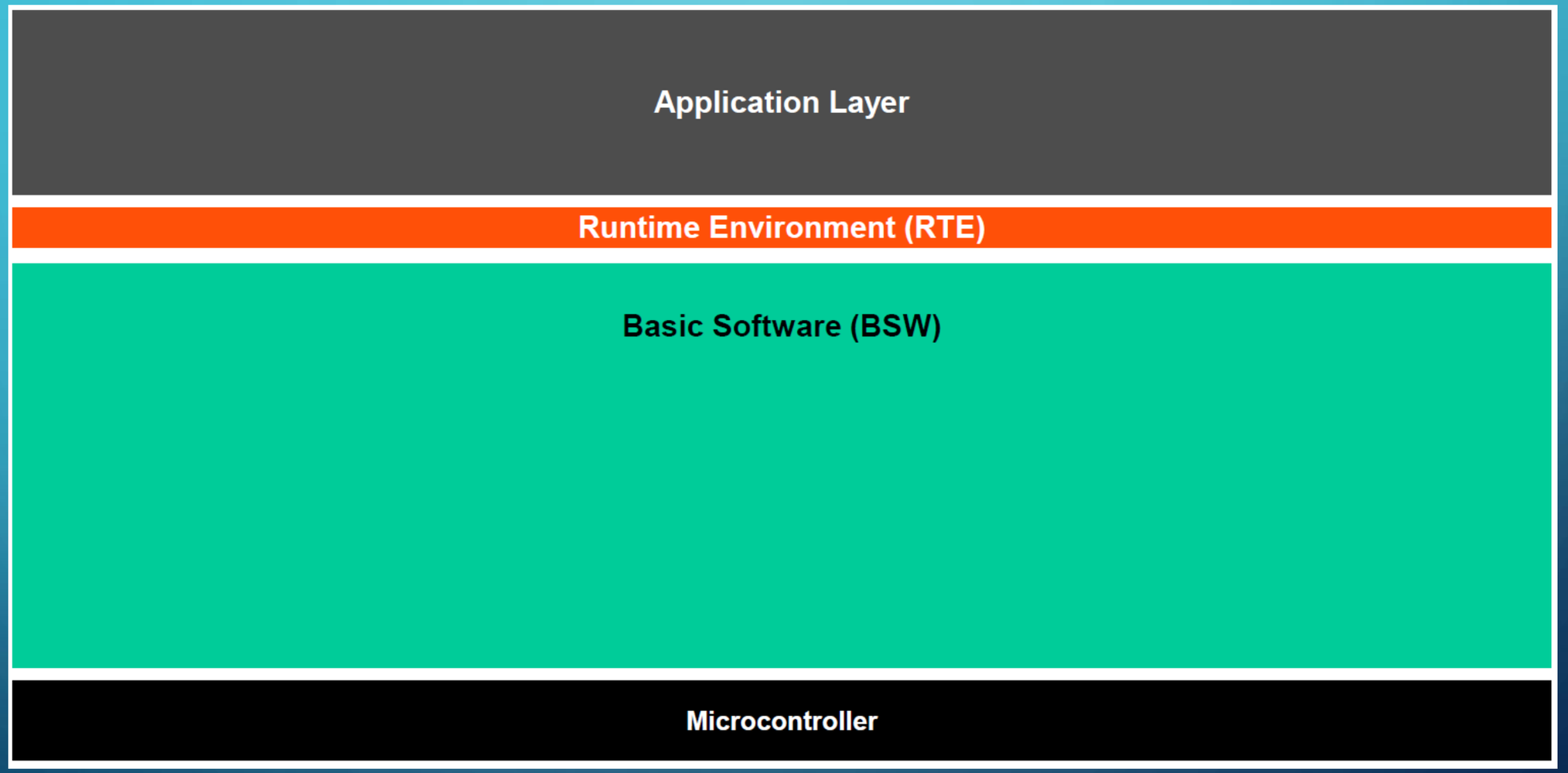
Exchange Formats



Methodology

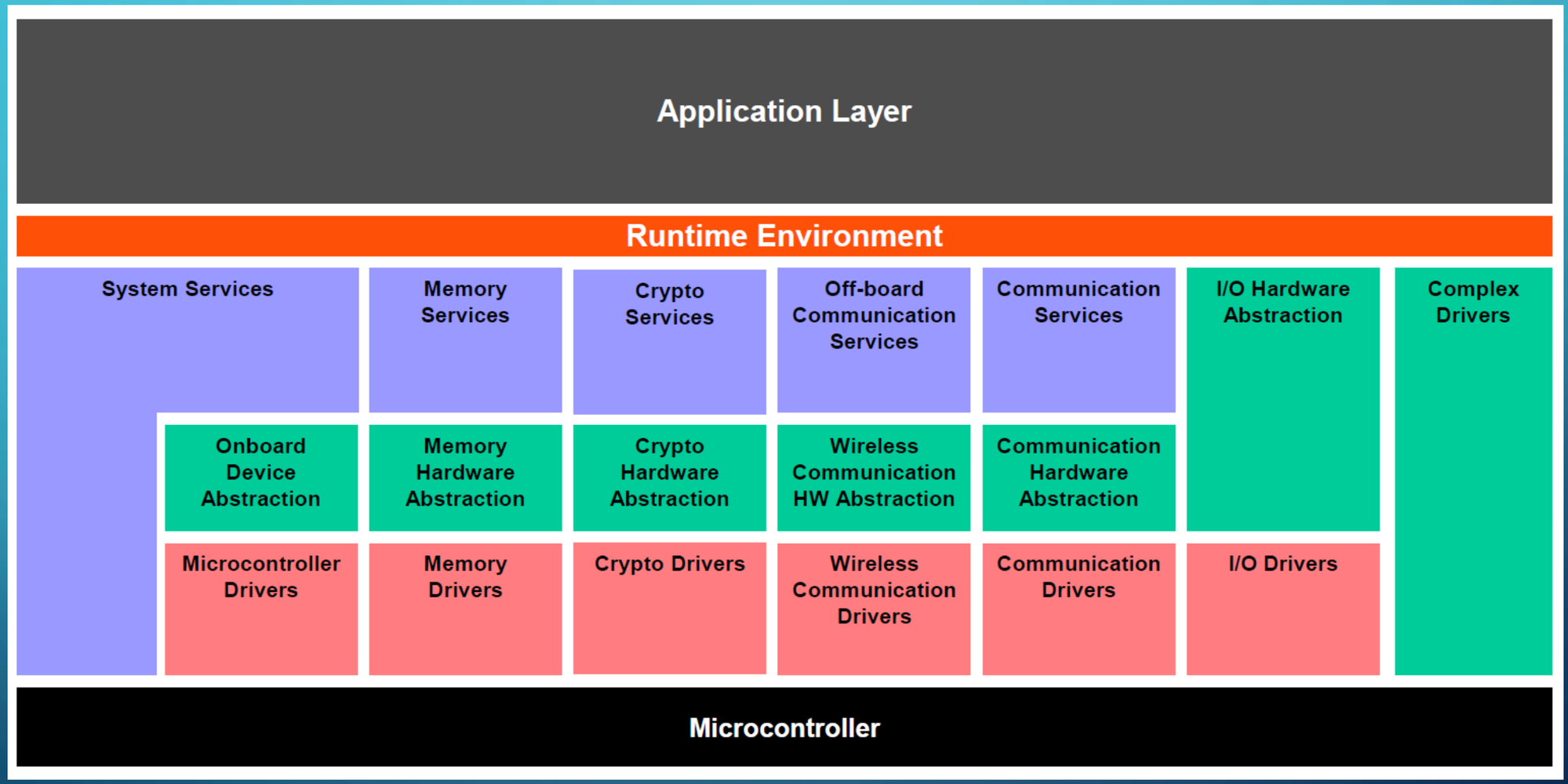
2KSYS s.r.o.

ARCHITECTURE



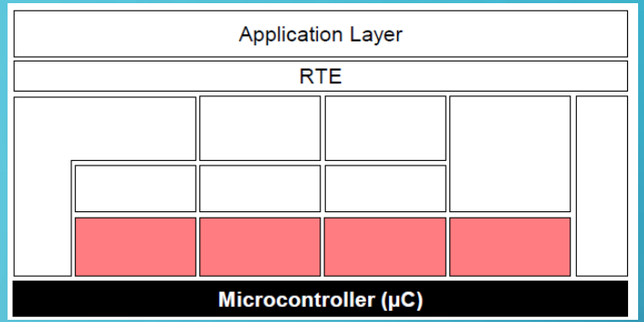
Source: AUTOSAR_EXP_LayeredSoftwareArchitecture 4.4.0

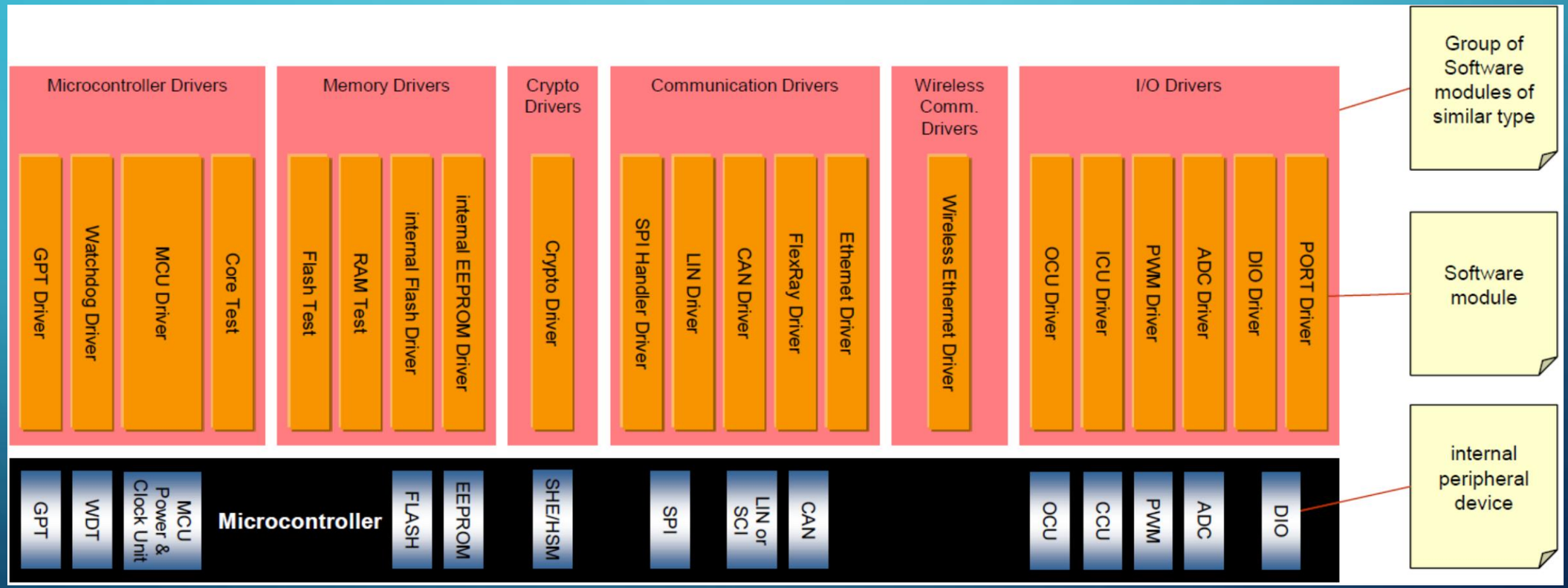
ARCHITECTURE



Source: AUTOSAR_EXP_LayeredSoftwareArchitecture 4.4.0

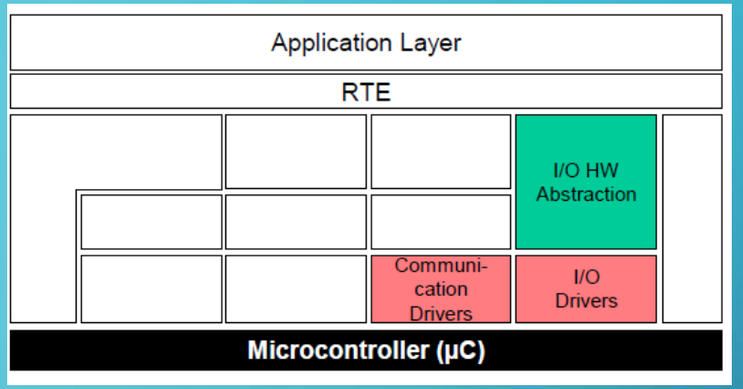
ARCHITECTURE - MCAL

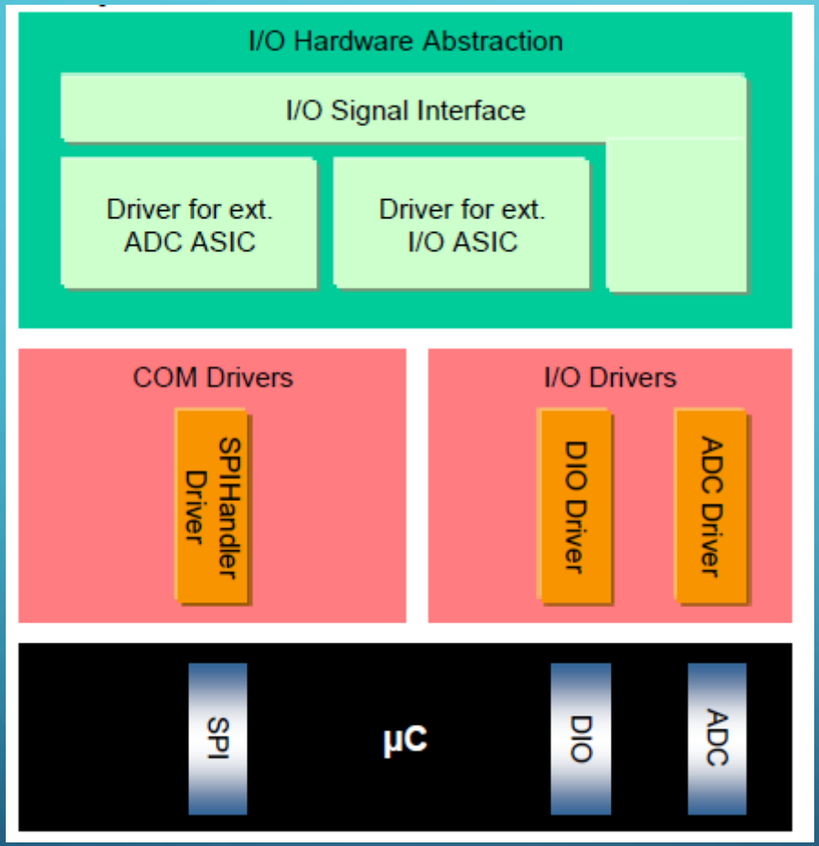




Source: AUTOSAR_EXP_LayeredSoftwareArchitecture 4.4.0

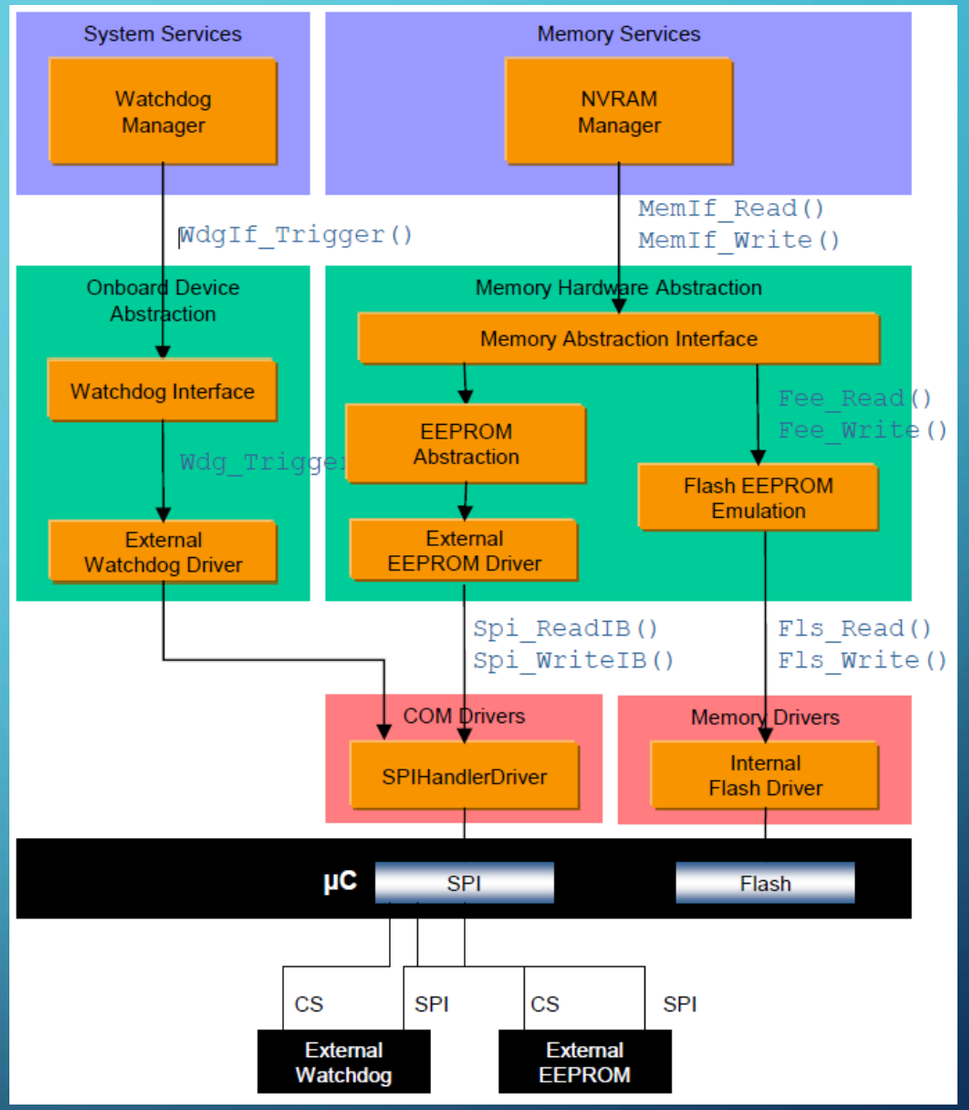
ARCHITECTURE – I/O HW AB





Source: AUTOSAR_EXP_LayeredSoftwareArchitecture 4.4.0

ARCHITECTURE – INTERACTION OF LAYERS



Source: AUTOSAR_EXP_LayeredSoftwareArchitecture 4.4.0

SAFE, RELIABLE AND ABOVE ALL FUN. IT'S TIME TO ENJOY!

2KSYS s.r.o.

THANK YOU FOR YOUR ATTENTION

JAROSLAV KADLEC

2KSYS

CZECH REPUBLIC

KADLEC@2KSYS.CZ

2KSYS s.r.o.