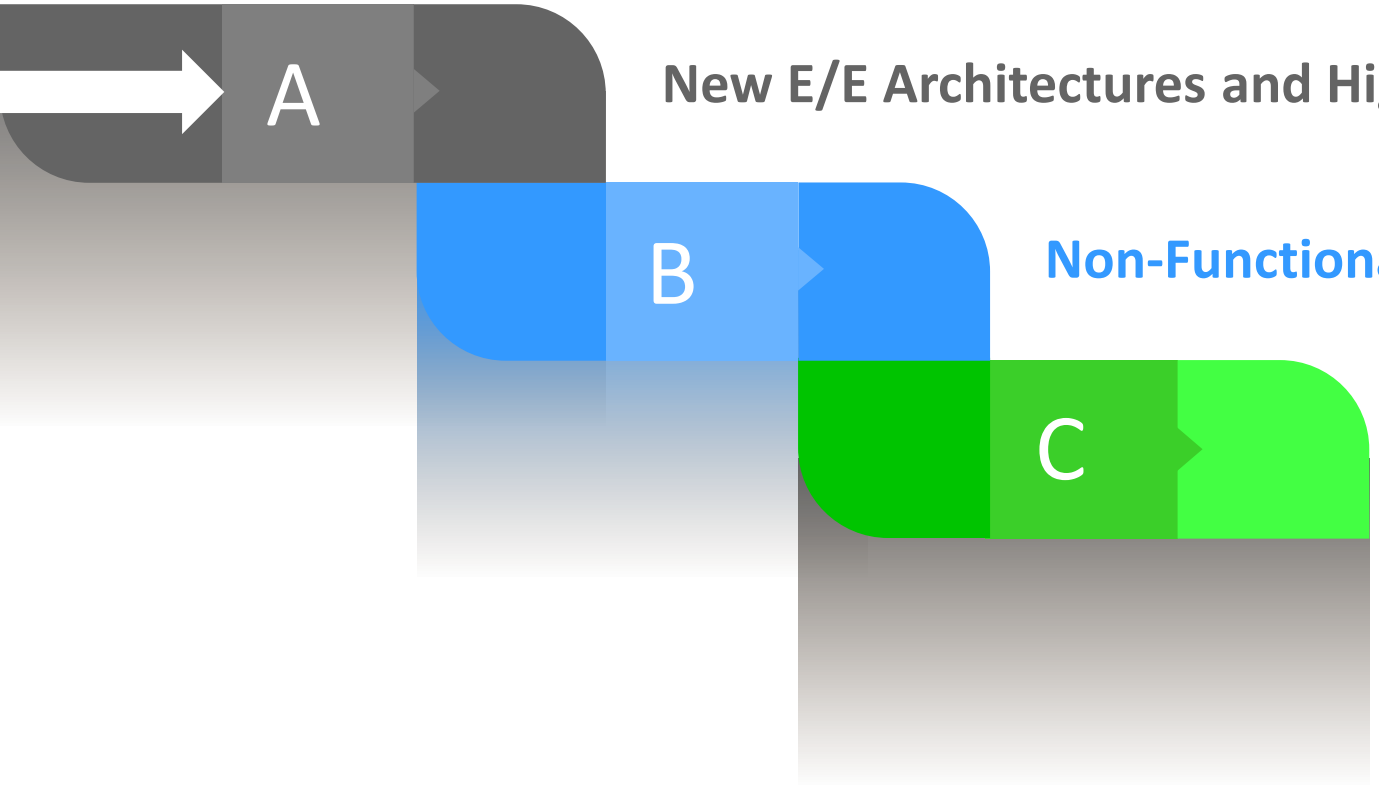
Scalable and Flexible Software Platforms for High-Performance ECUs

Christoph Dietachmayr
Sr. Engineering Manager, Elektrobit
November 8, 2018

Agenda



New E/E Architectures and High-Performance ECUs

Non-Functional Aspects: Safety | Security | Cloud

Adaptive AUTOSAR / EB corbos:
Where Safety Meets Performance

Major Market Trends

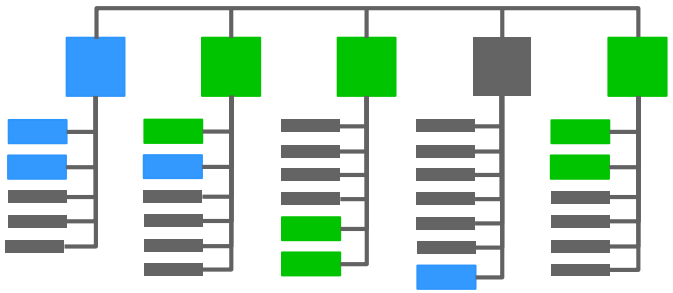
Trends		Impact on E/E Architecture	Impact on SW Architecture
	E-Mobility	Shrink of powertrain reduces hw complexity	• Software complexity increase • Central car computer approach
	Automated Driving	High data volumes Safety meets performance	• Fail operational systems • Availability • Service oriented architecture (SOA)
	Mobility Service	Car-to-X connectivity Update and Upgrade capability (over the air)	• Holistic security approach • Dependability

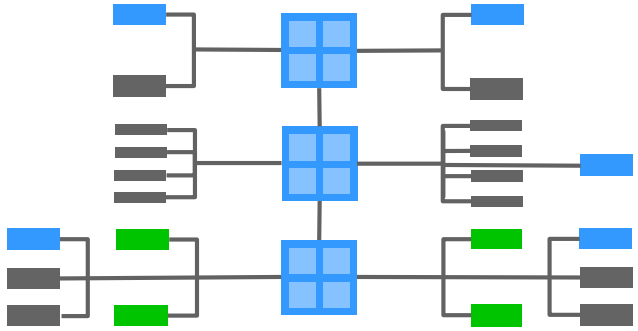
Adaptive AUTOSAR
base technology for

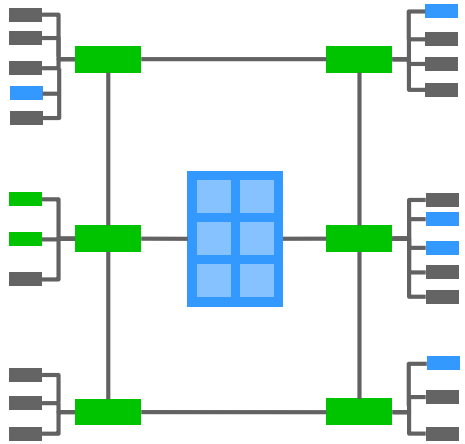
- safe
- secure
- flexible
- up to date
- high performant

central car control units.

Evolution of E/E Architectures







Today

- Signal based communication
- System of ECUs
- Predictable communication
- Function orientated topology

Tomorrow

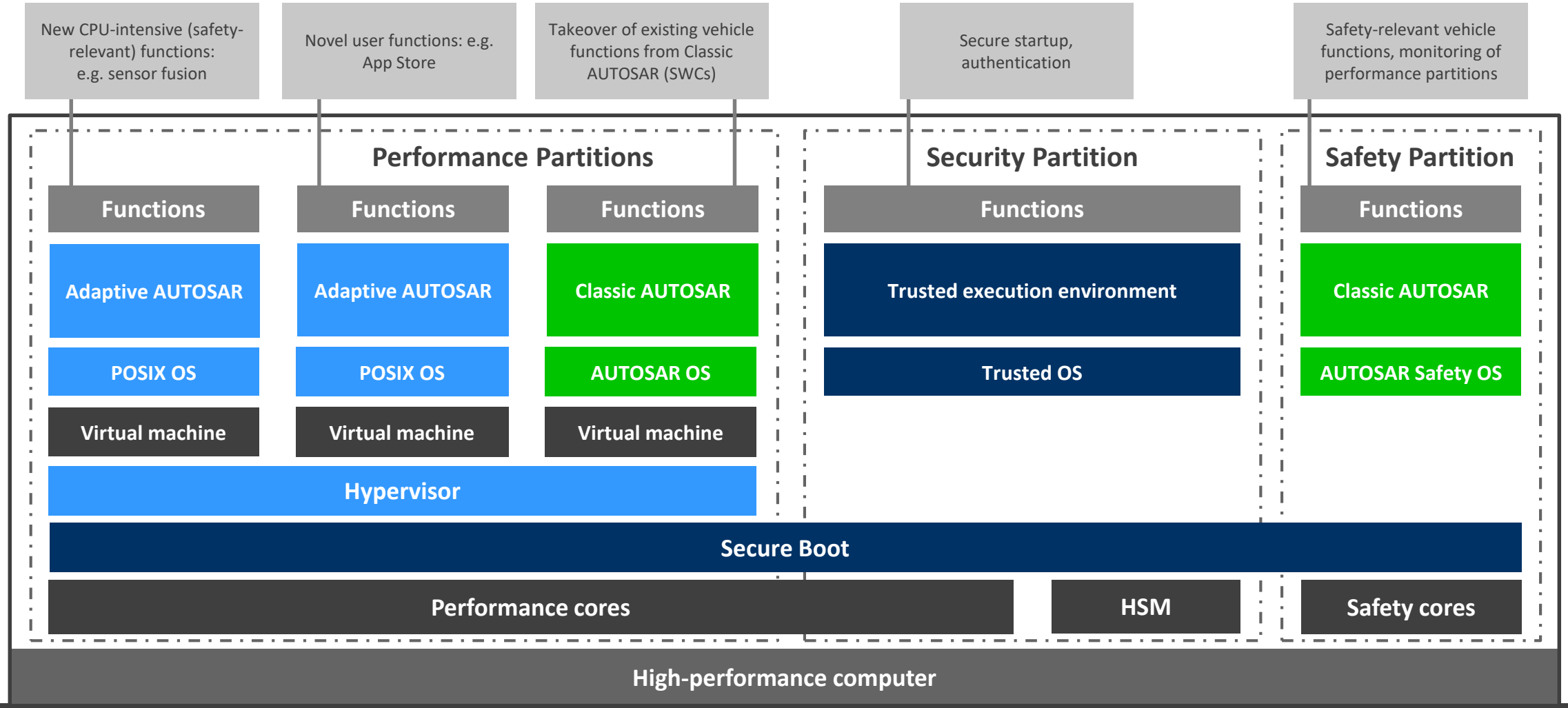
- Central computing nodes
- Mix of signal based and service orientated communication
- Partly centralized functions
- Software upgradability

Future

- IP/Ethernet communication
- Centralized applications/functions
- Computing power for HAD and AI
- Anything anywhere (sensors/actors)

Potential HPC Architecture – SOP 2019

Infrastructure Software (Operating System and Middleware)

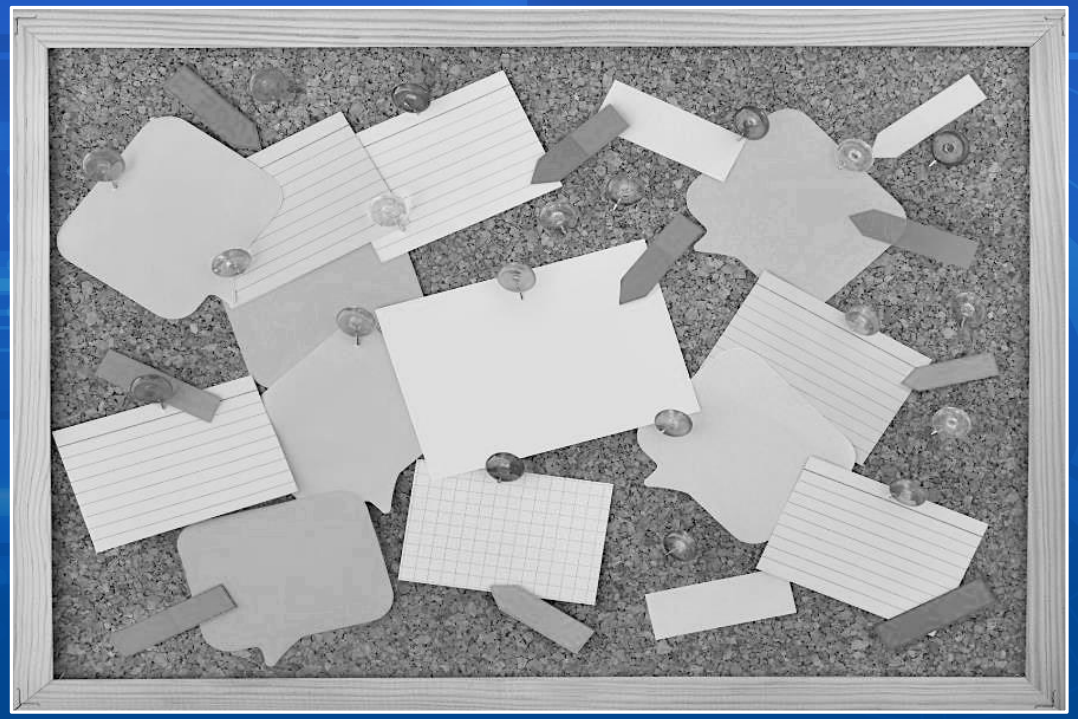


Elektrobit

Communication in a Service-Oriented Architecture

Public Speech

Bulletin Board



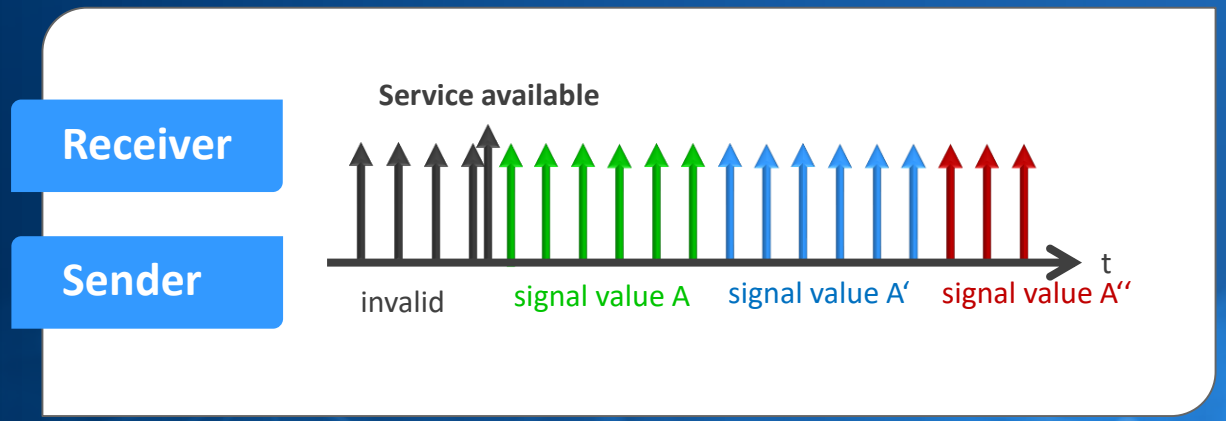
Signal Based vs. Service Based Communication

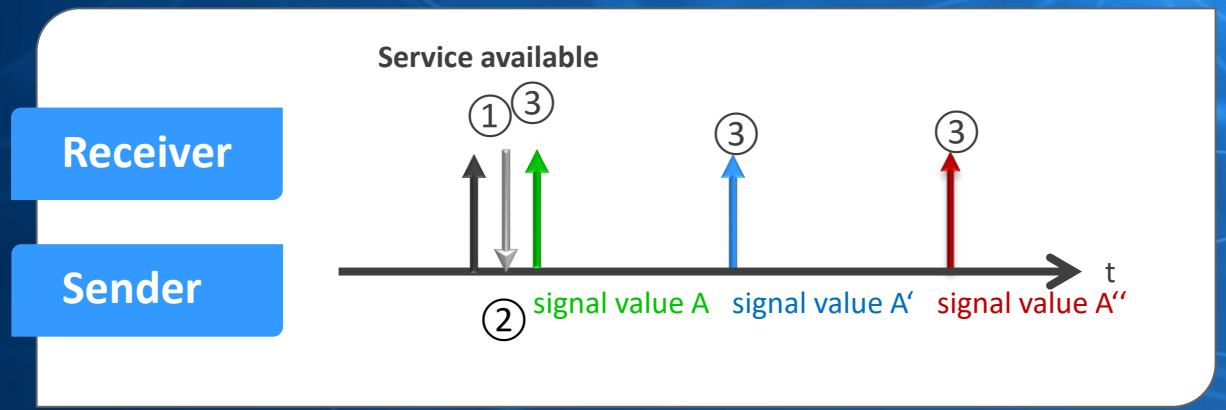
Signal based

- Sender
  - Just transmit (implicit availability info)
  - Doesn't matter whose receiving
- Receiver
  - Just Listen
  - Doesn't matter whose sending

Service Based

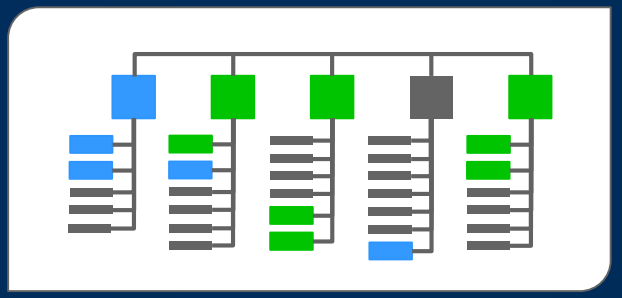
- Sender (= Service Provider)
  - Offer service & maintain subscriber ①
  - Send message to subscriber
- Receiver (= Client)
  - Find service & subscribe to service ②
  - Receive message ③

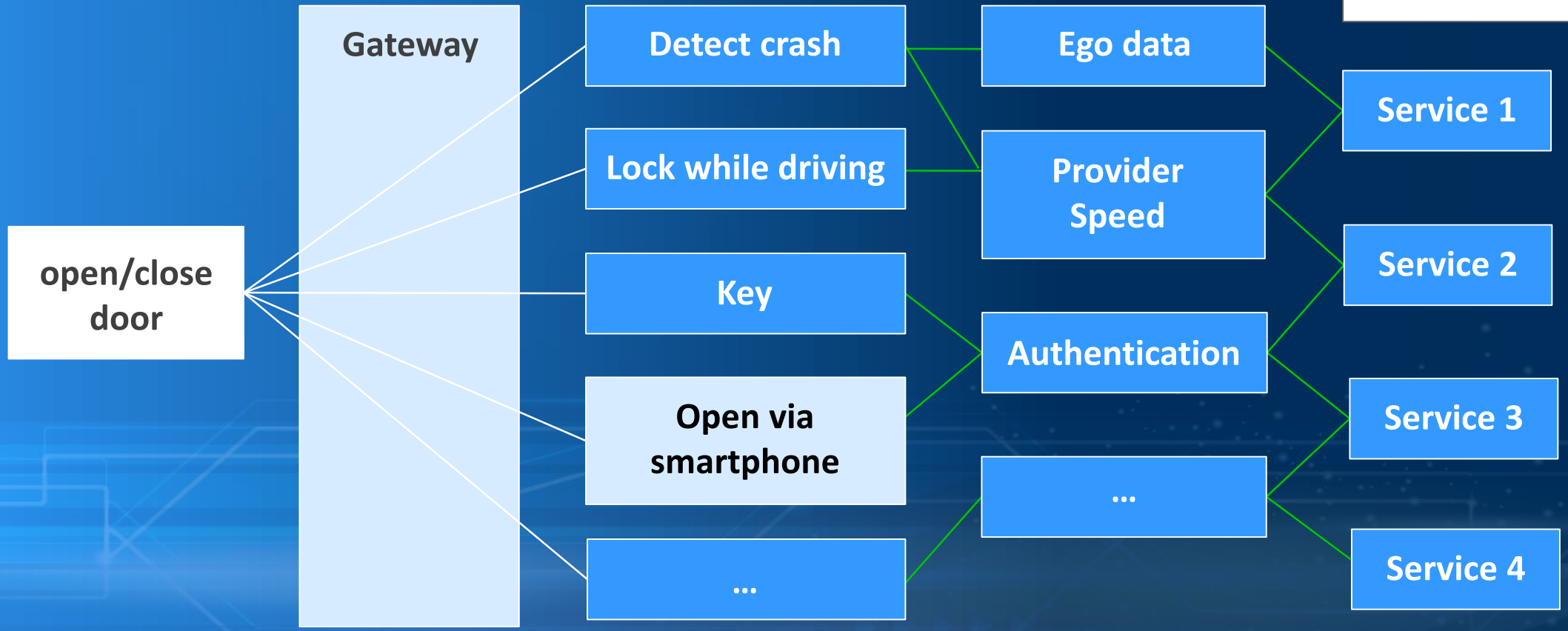




Elektrobit

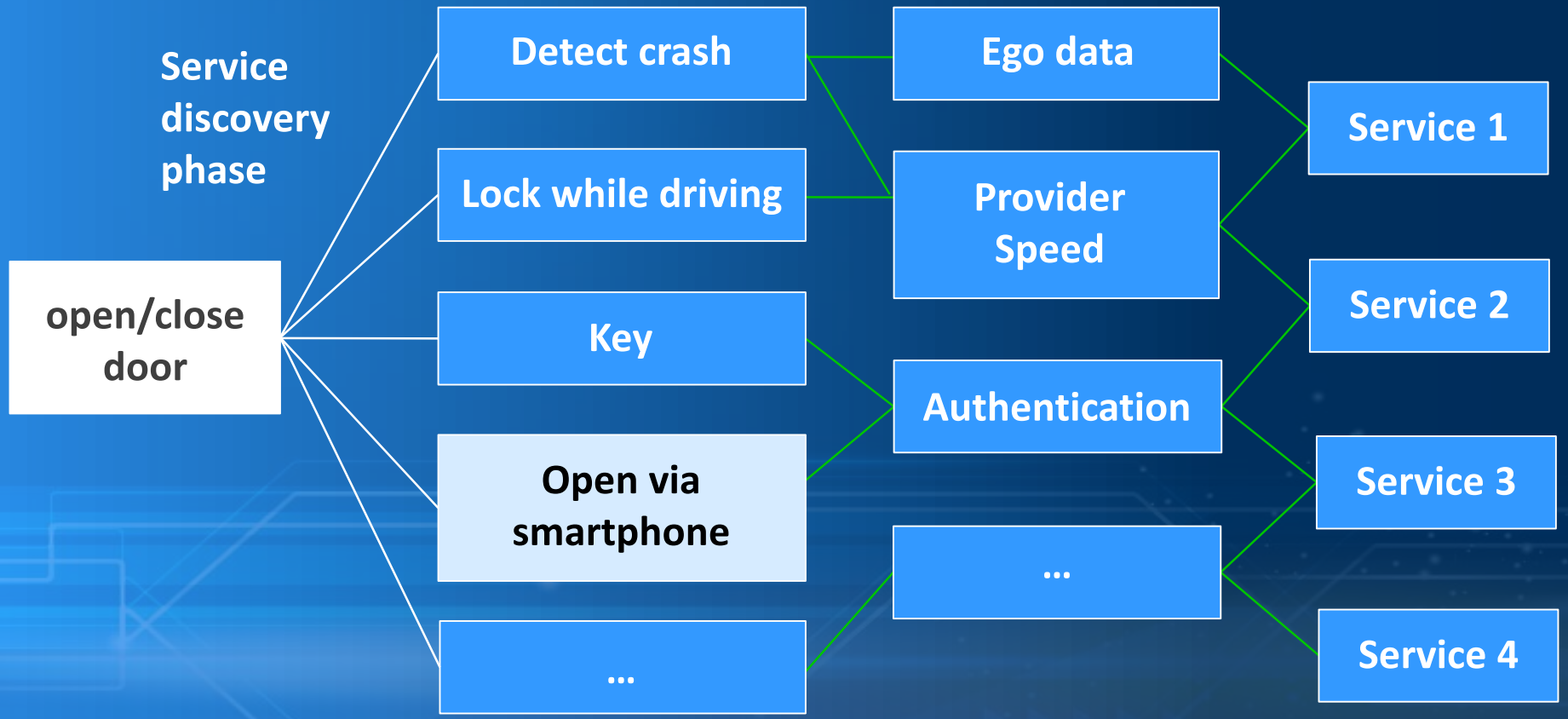
Signal Based: New Services Require Changes to Gateway

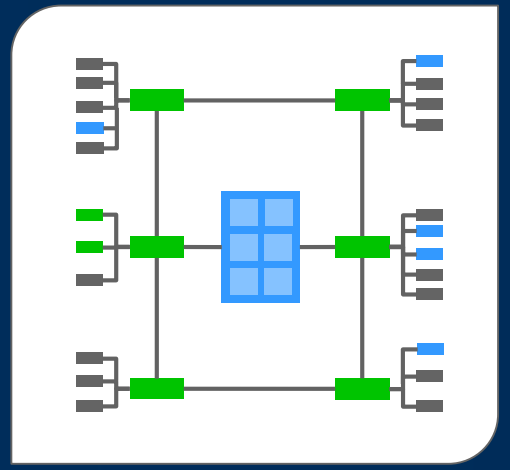




Elektrobit

Service Discovery Phase – Find & Match Services

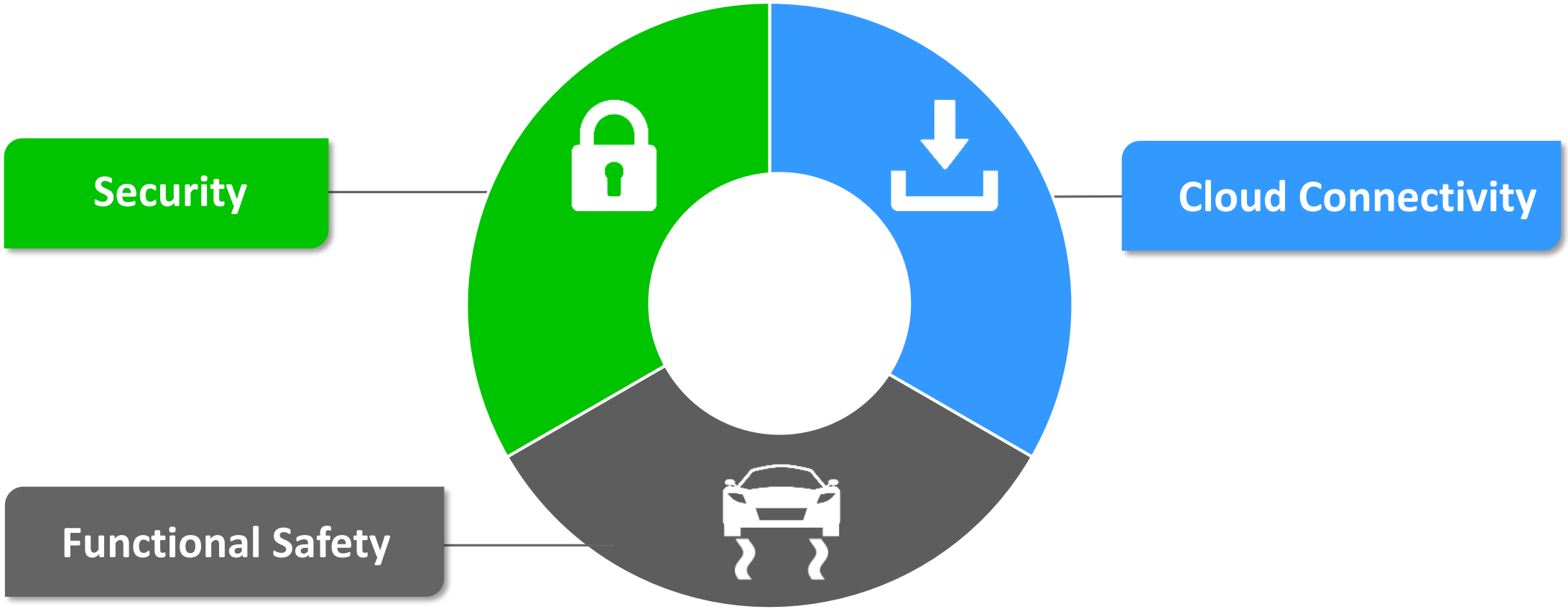




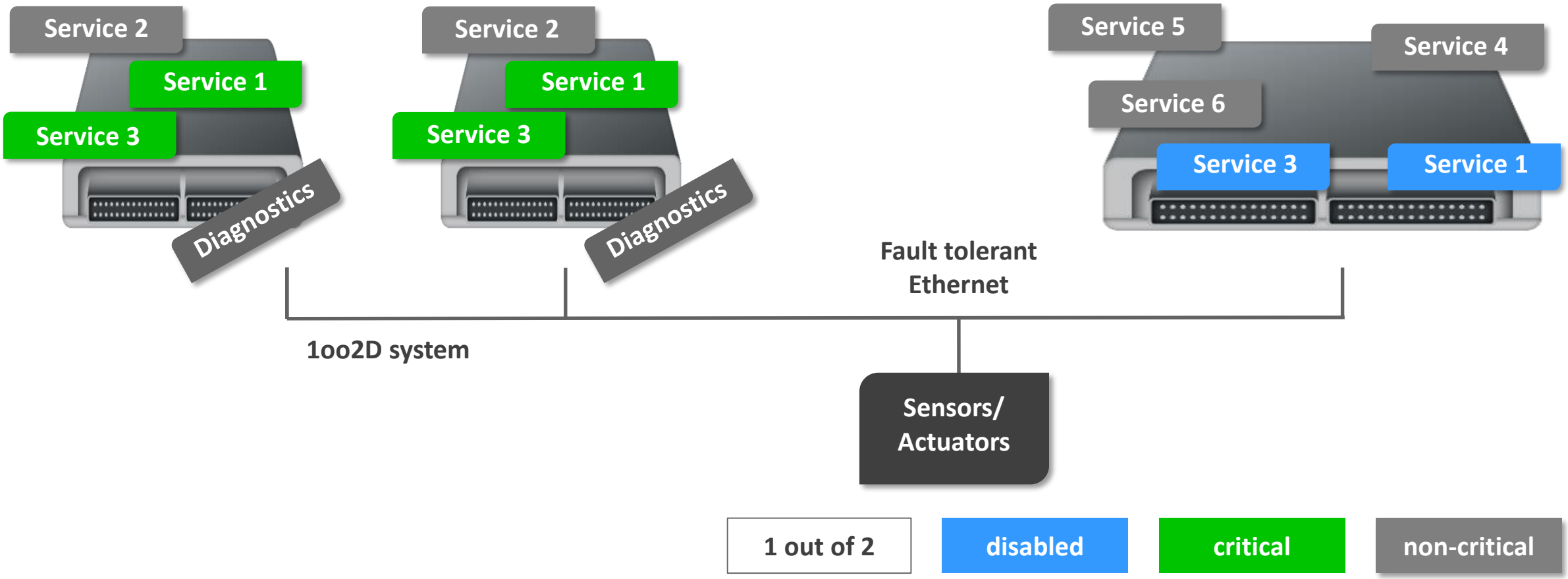
Services can be distributed - flexible on various ECUs

No dependency or changes in Gateway necessary

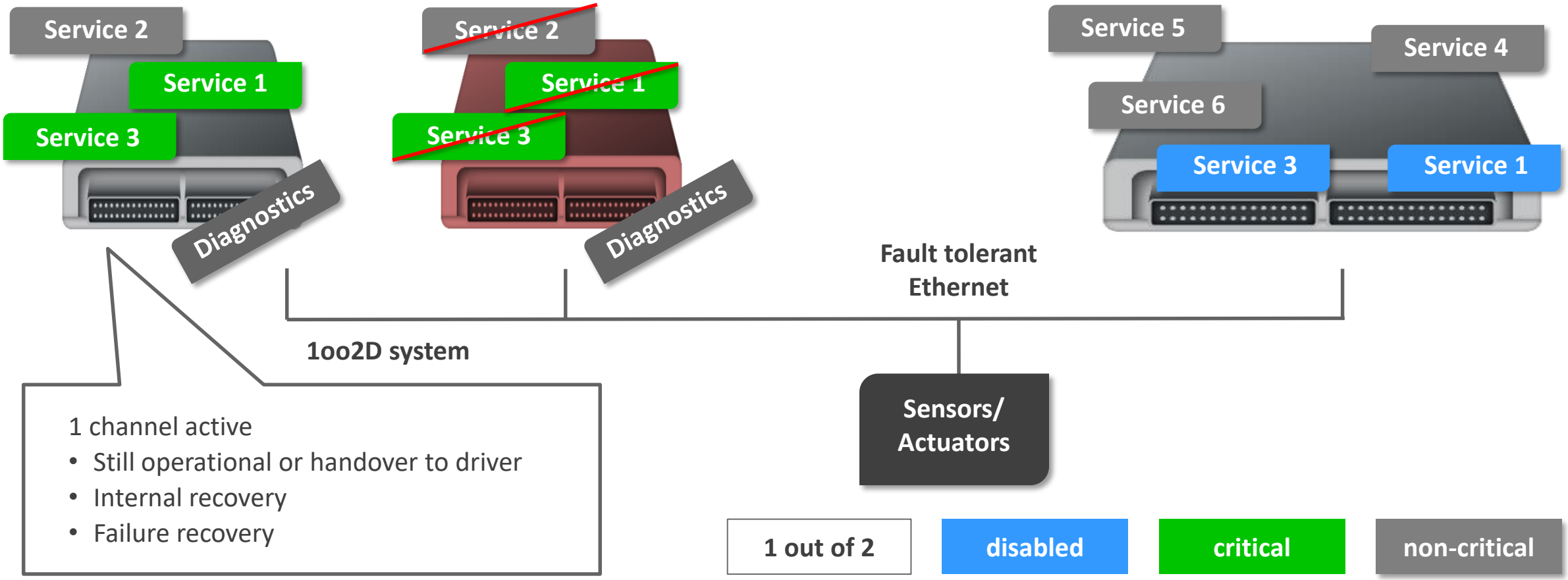
Additional Non-Functional Requirements Arise



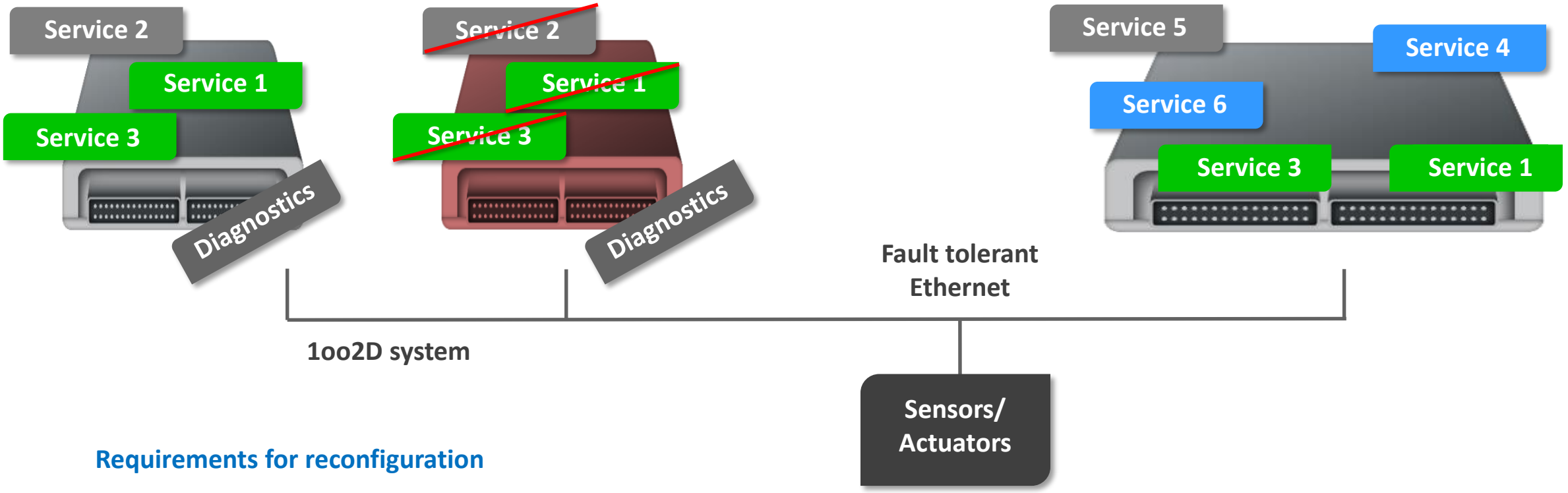
Normal Operation of Distributed Service



One Channel Failed



Reconfiguration of Services



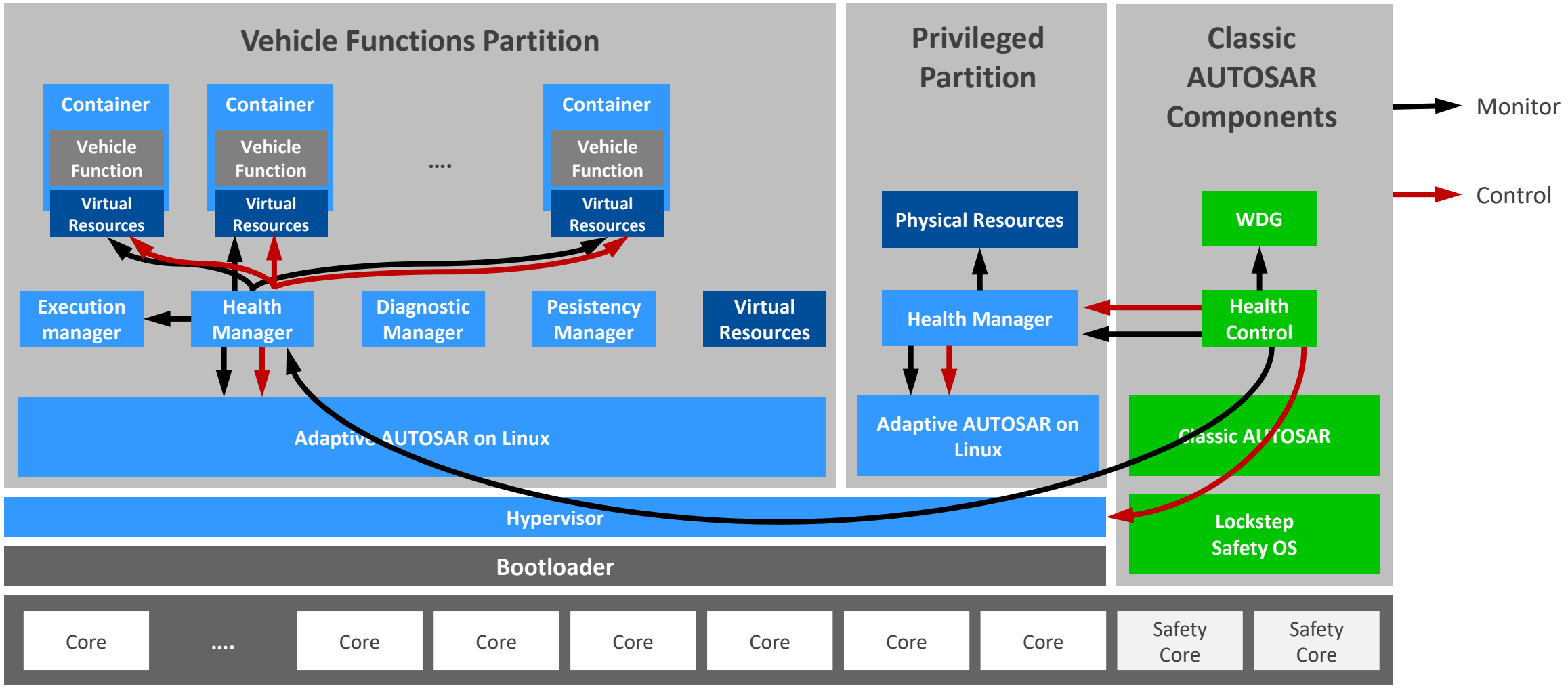
Requirements for reconfiguration

Req. 1: Services can be dynamically relocated

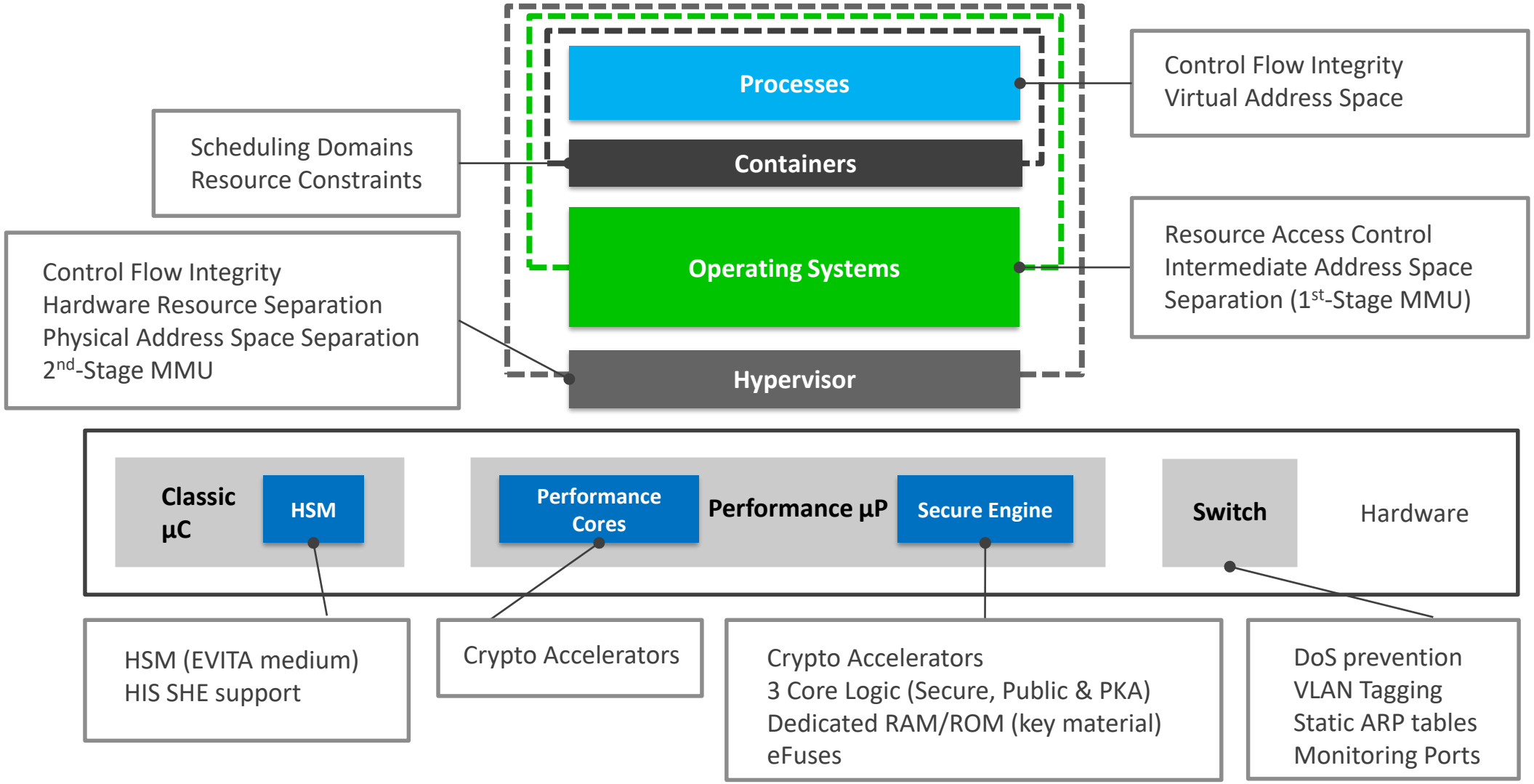
Req. 2: Sensor/actuators are redundant or accessible via network as a service

1 out of 2 | disabled | critical | non-critical

Safety Monitoring Concept



Security Layer Concept



Cloud Connectivity: Use-Case Remote Update

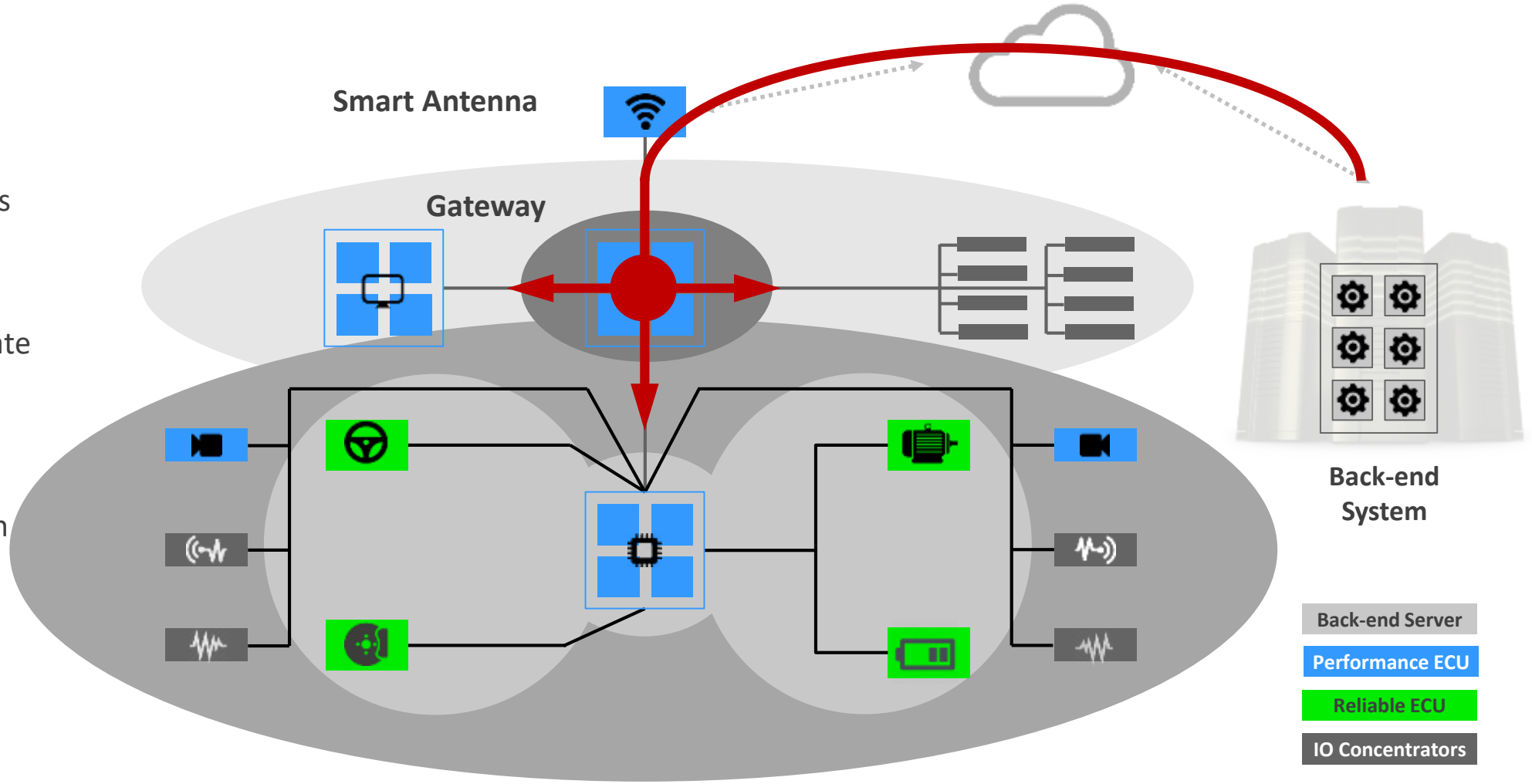
Architectural Principles:

- Central external connection
- Distribution of updates across multiple ECUs

Supporting Features:

- Coordinated A/B Update across ECUs
- Application containerization
- Secure communication
- Layered security architecture

Company infrastructure and IT systems to be included



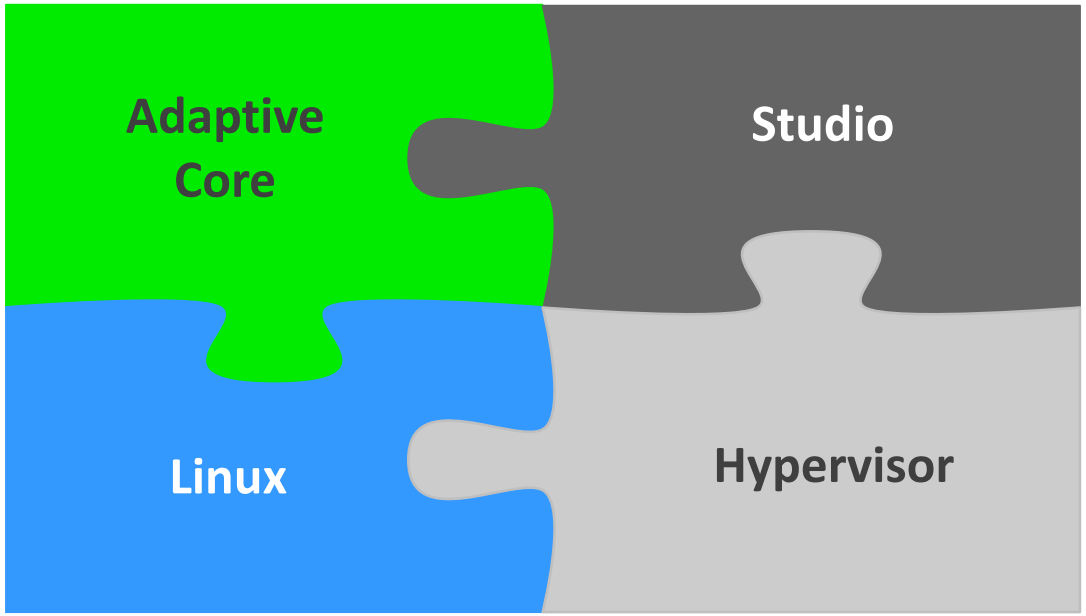
Adaptive AUTOSAR – EB corbos Where Safety Meets Performance

EB corbos AdaptiveCore

AUTOSAR Adaptive Platform for safe and secure applications

EB corbos Linux

Container based Linux



EB corbos Studio

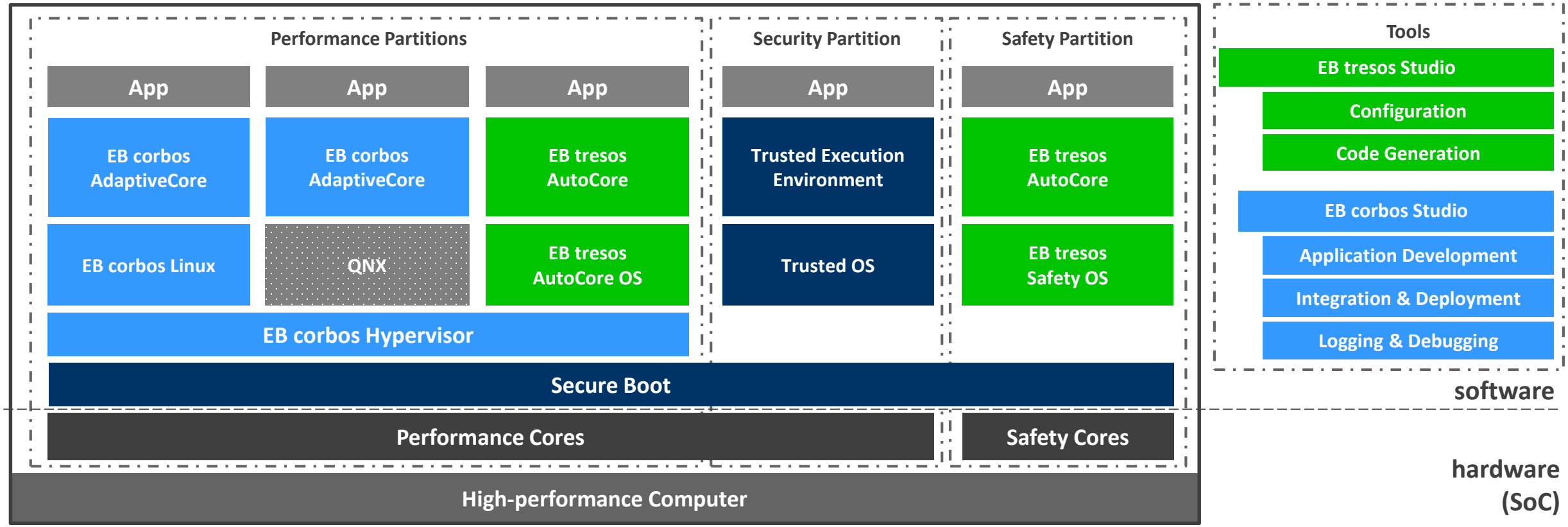
Integrated development environment for EB corbos products

EB corbos Hypervisor

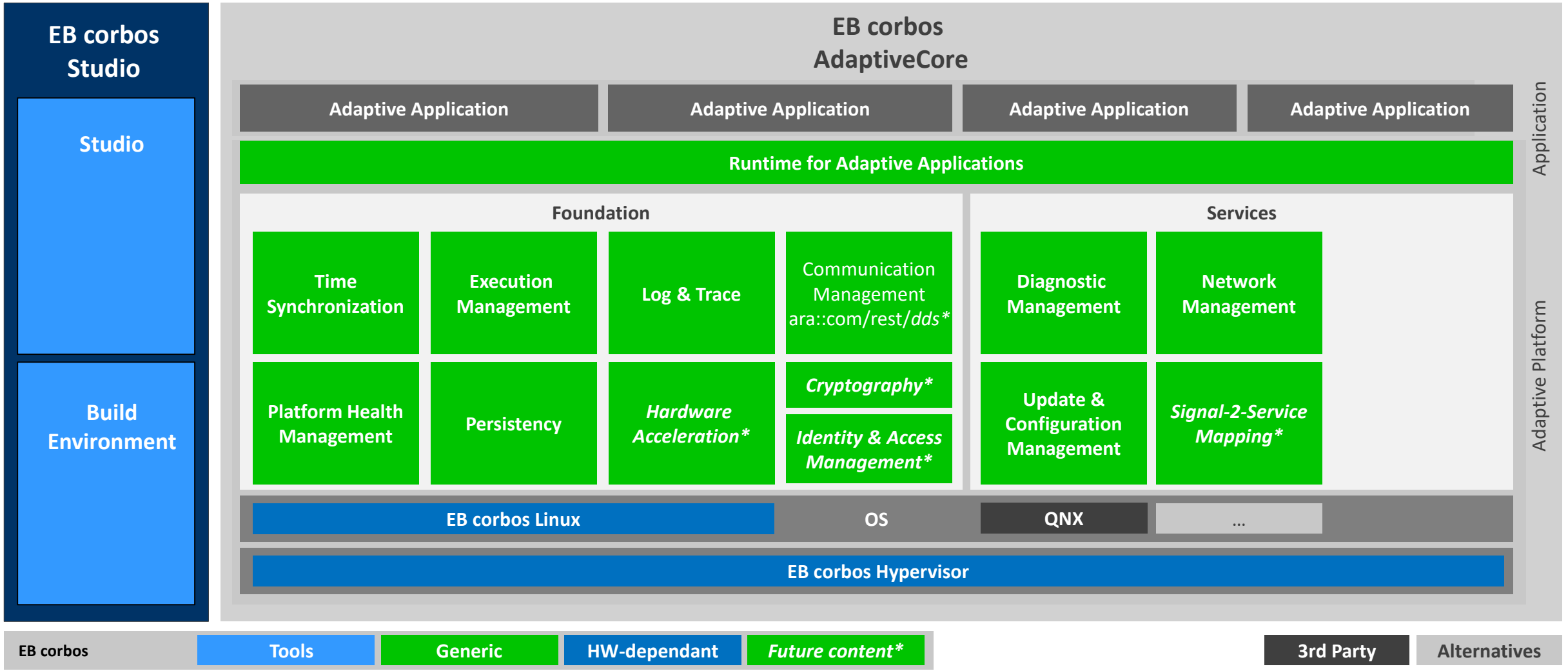
Micro-Kernel based type1 hypervisor to run multiple (different) operating systems on one single CPU

EB's High-Performance Computer (HPC) offering…

One-Stop Shop Software Provider for Next Generation ECUs



EB corbos AdaptiveCore



EB corbos Studio

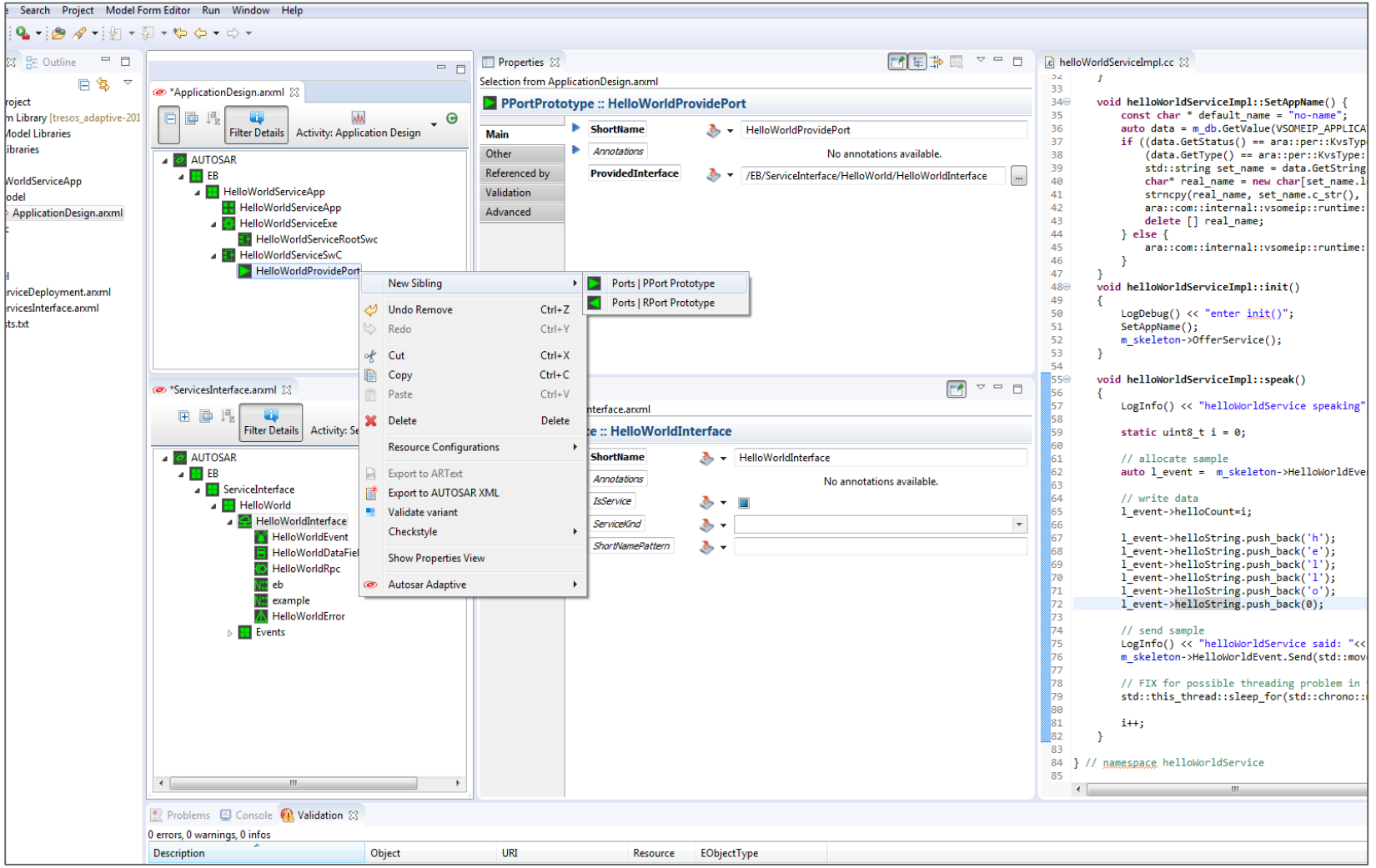
Integrated Development Environment for EB corbos Products

- **Application Development**

  Develop applications according the AUTOSAR Adaptive platform.

- **Integration**

  Integrate software components with EB corbos AdaptiveCore and other products from EB corbos product portfolio.



EB corbos Studio – Features

Application development

Create and modify application design

- Specify the application for the AUTOSAR Adaptive platform
- Add and modify required and provided services
- Use system data model usually provided by the OEM

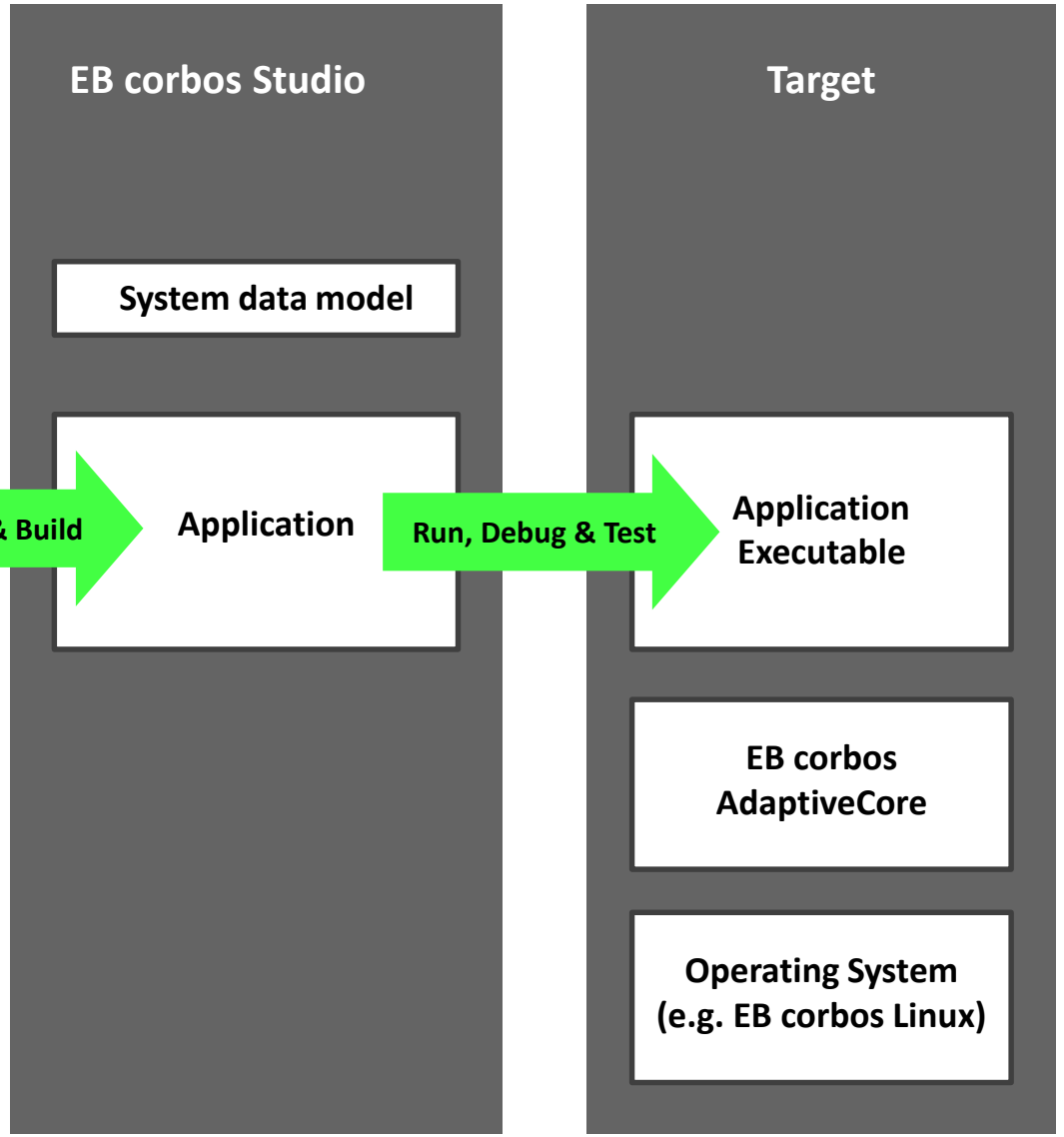
Implement and build the application

- Implement the application in C or C++

Run, Debug and Test the application

- Run and debug the application on the PC in a virtual machine running EB corbos Linux, or directly on the target

Application Developer



EB corbos Studio – Features

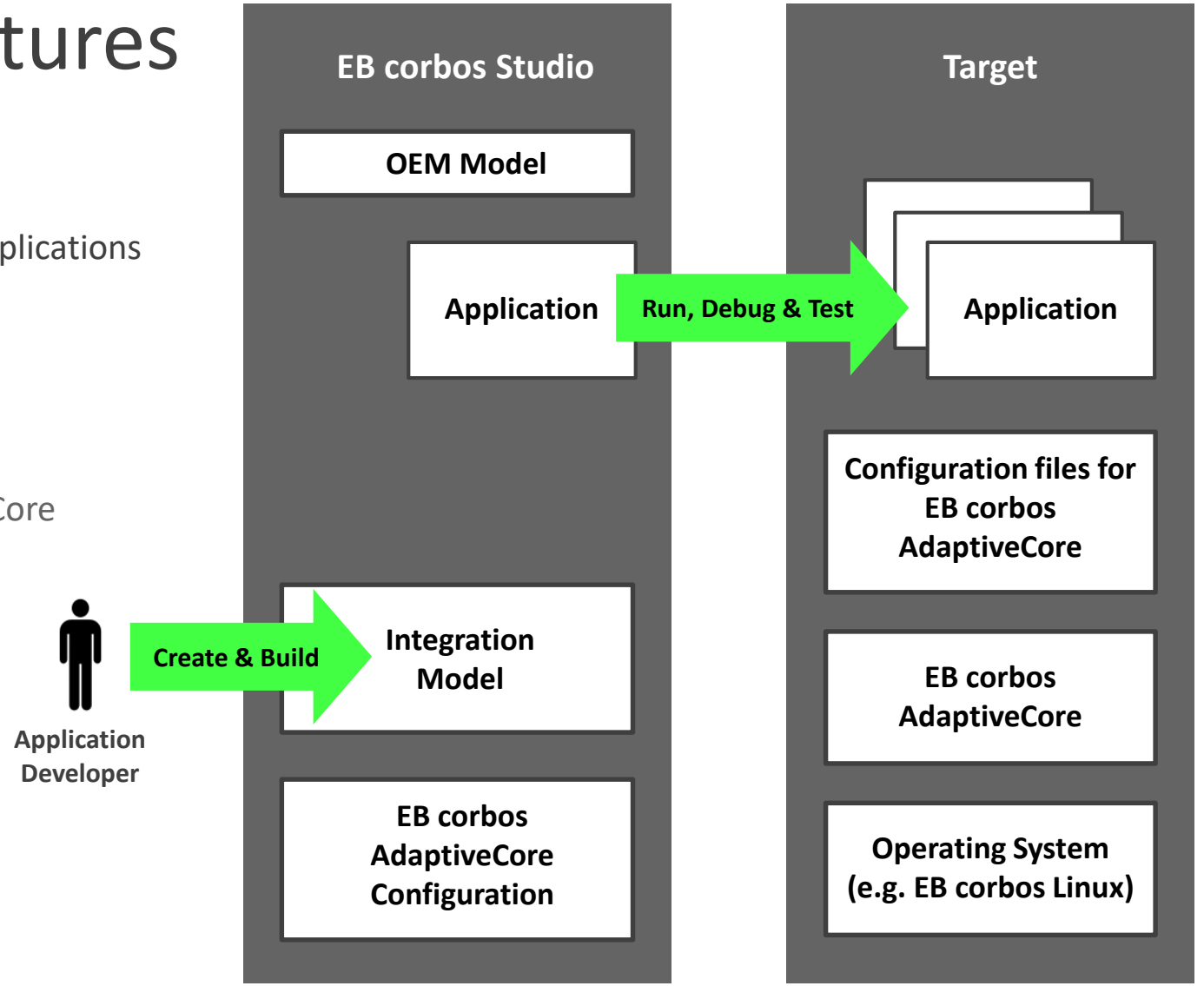
Integration

Integrate applications

- Define process and startup parameters for the applications
- Define dependencies to other applications

ECU integration

- Create configuration files for EB corbos AdaptiveCore
- Configure network connectivity
- Define ECU and vehicle states



Conclusion

- **Define Performance controllers build on top of a software system architecture on the basis of AUTOSAR**
- **Adaptive AUTOSAR alone is not the solution for all performance controllers**
- **Classic and Adaptive AUTOSAR form a foundation for complex automotive software systems**
- **System properties must be ensured thru system architecture, particularly for Safety / Security / Reliability**

Thank you for your attention!

Questions?

Christoph Dietachmayr
Sr. Engineering Manager, Elektrobit
Christoph.Dietachmayr@elektrobit.com