KERNKONZEPT

SAFE AND SECURE HETEROGENEOUS COMPUTE WITH THE L4RE HYPERVISOR FAMILY

Dr.-Ing. Adam Lackorzynski | CTO - Kernkonzept | TU Dresden

2023-05-31 Huawei GSTS, Dresden

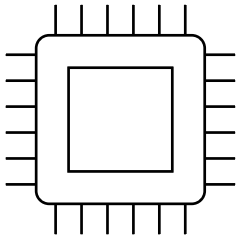
KERNKONZEPT

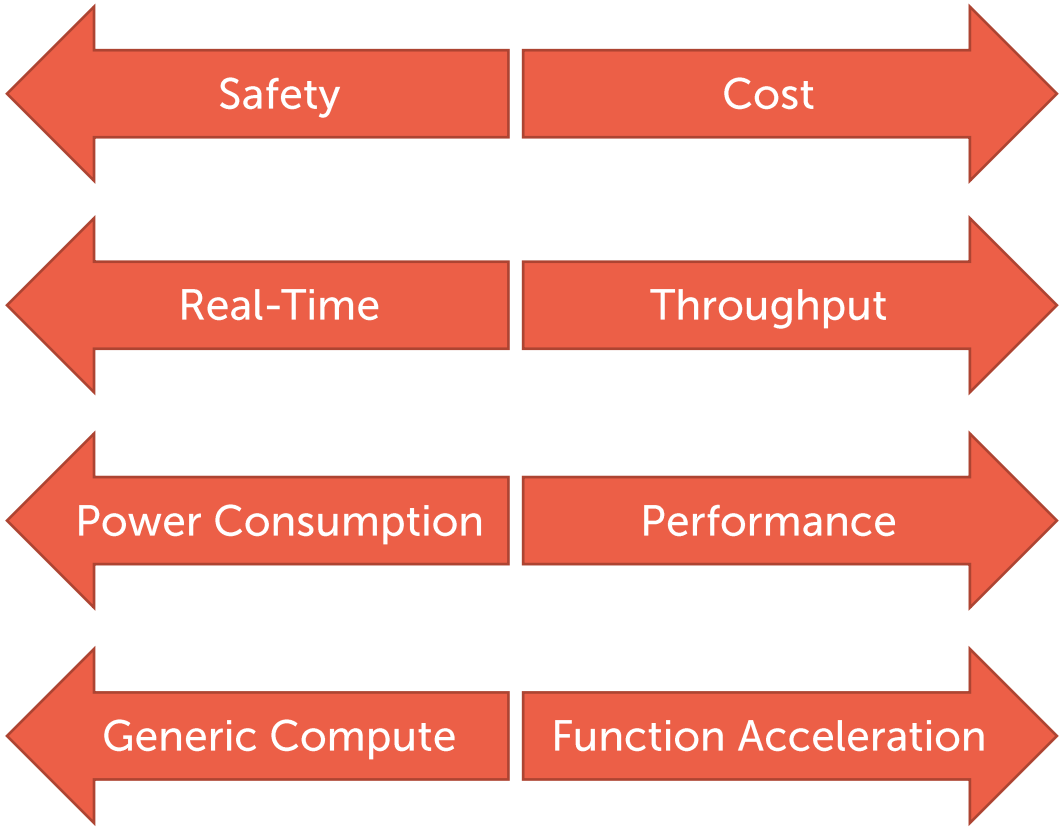
Trend: Consolidation & Feature Demand

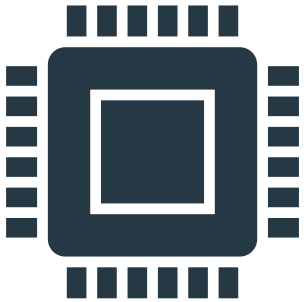
+ **Driver for Software Defined Vehicle (SDV) in Automotive**

+ **Consolidation: Put more software on less hardware**
  - Hardware has to cover a wide range of functionality

Challenges in Hardware Design



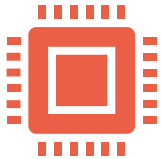




+ **Processors are optimized for different targets**
+ **Diverse Hardware Design**

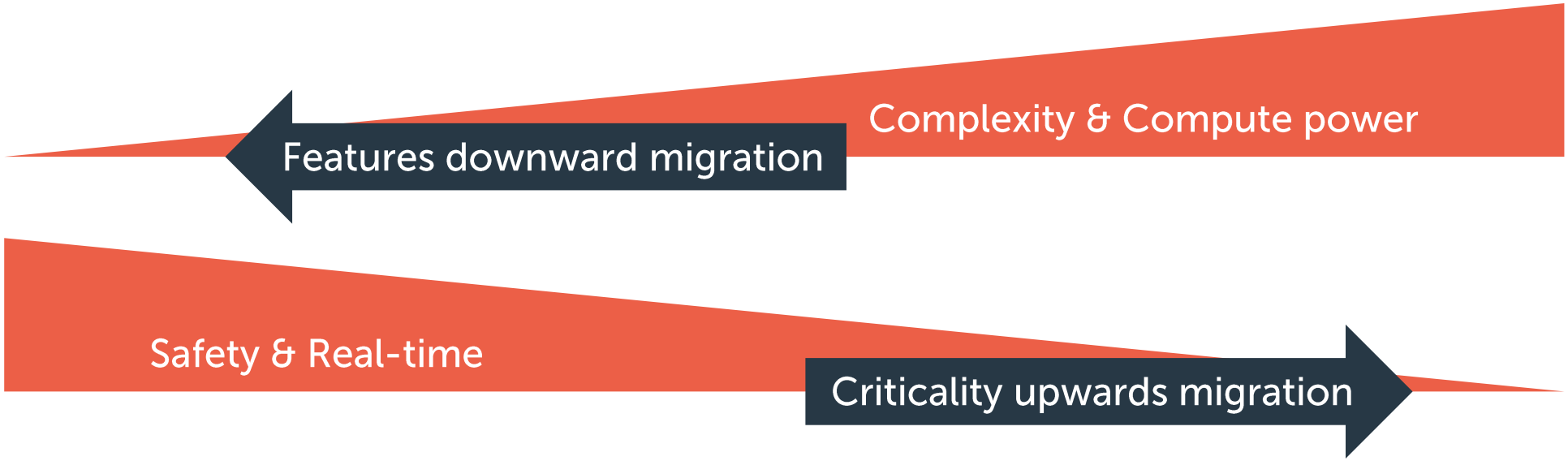
+ **Heterogenous Hardware**

Range of Hardware Diversification



Small-scale systems,
MPU-based protection,
e.g.: Arm Cortex-R

Big-scale systems,
MMU-based protection,
e.g., Arm Cortex-A, x86



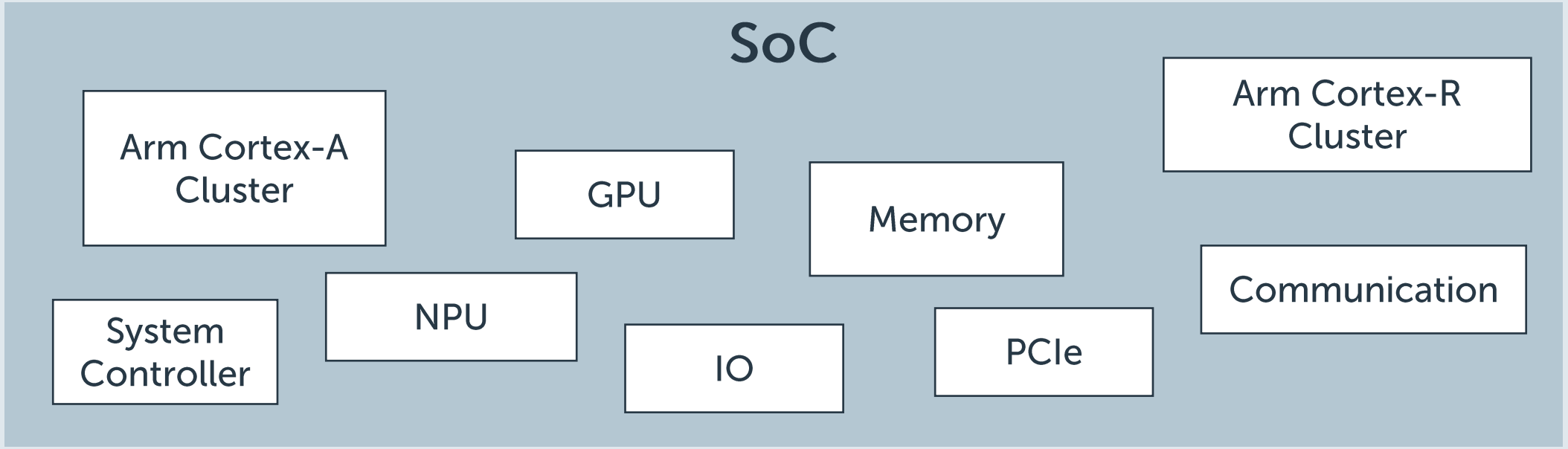
Convergence of Processor Designs

- **Processor designs gaining features, examples:**
  - Arm Cortex-A76AE: Enabled for safety critical environments
  - Arm Cortex-R52: Enabled for virtualization

- **Architectural differences remain:**
  - MMU vs. MPU
  - 64bit vs. 32bit
  - ISAs

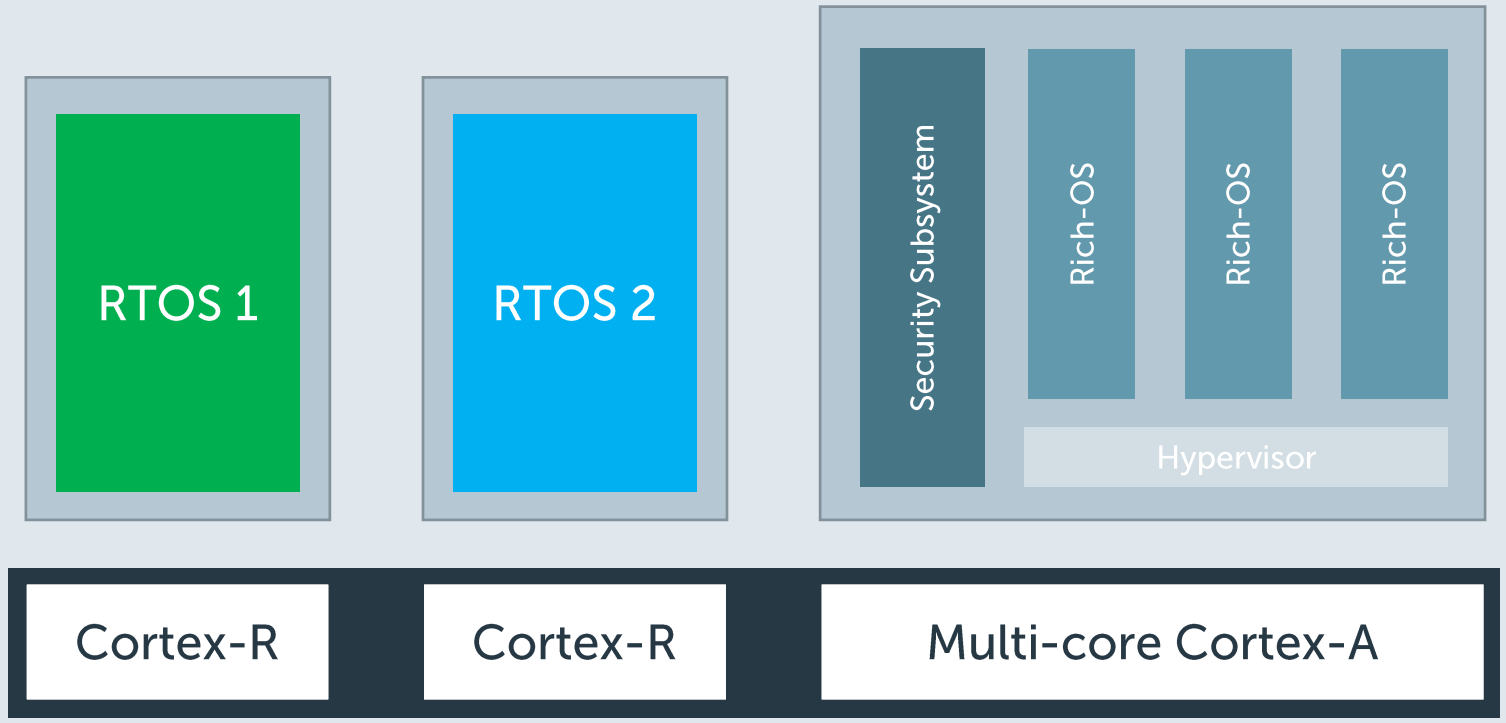
Heterogeneous Hardware

+ **Powerful SoCs are complex and a combination of several different compute units**



Heterogeneous Hardware and Software

+ **Typically, all compute units run their own different OS with their own application stack.**



The Software Defined Vehicle (SDV)

+ **Vehicles are defined through software**

+ **Software is the majority of effort and invest**
  - Must work over a wide range of
    - Use-cases
    - Vehicles variants
    - Vehicle generations

Software Independence

+ To Hardware:

- Flexibility to compose software components according to hardware performance, price and availability constraints

+ Between software components

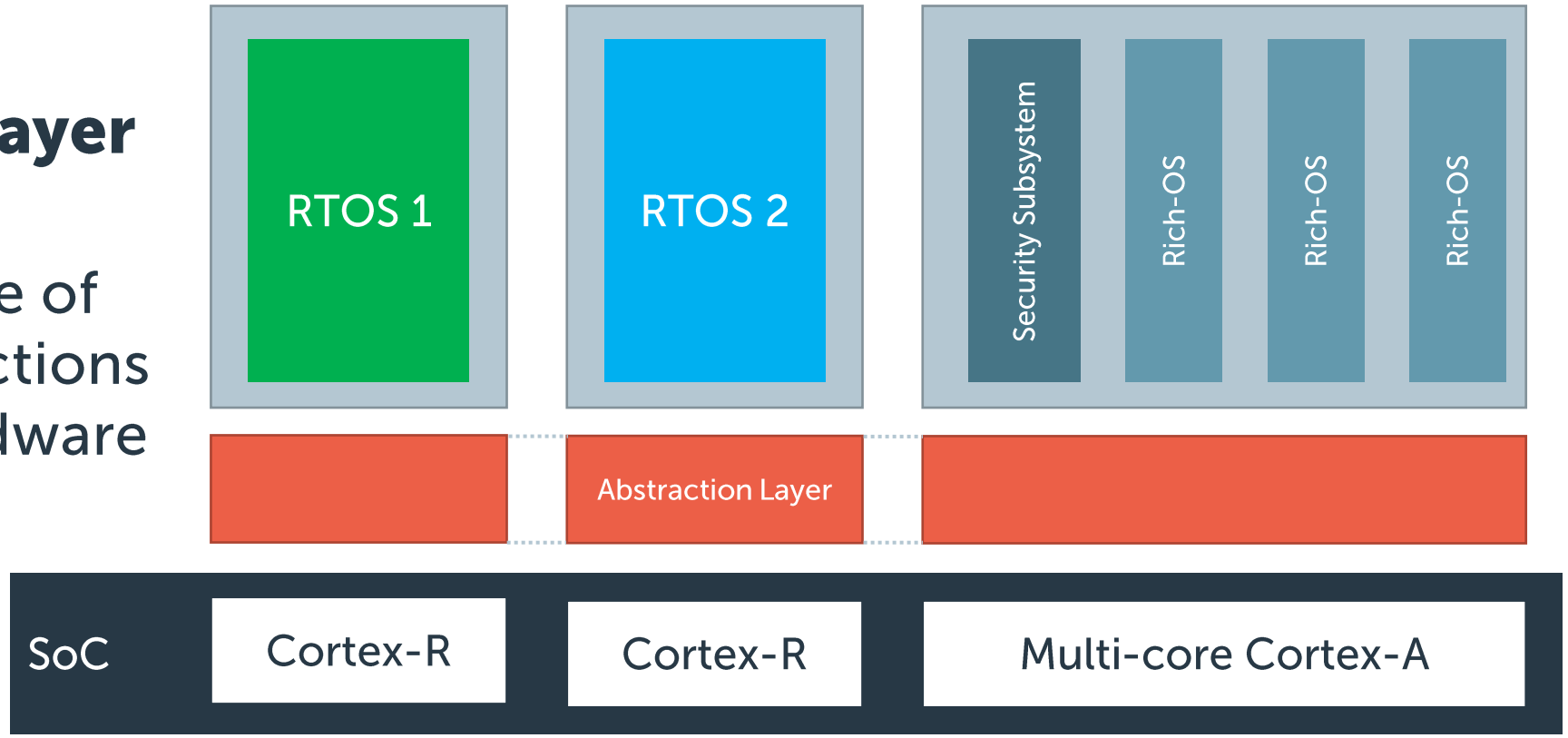
- Independent software life cycles
- Certification requirements of components should be decoupled

+ Independence → Updates

- Software & Hardware exchange / upgrade

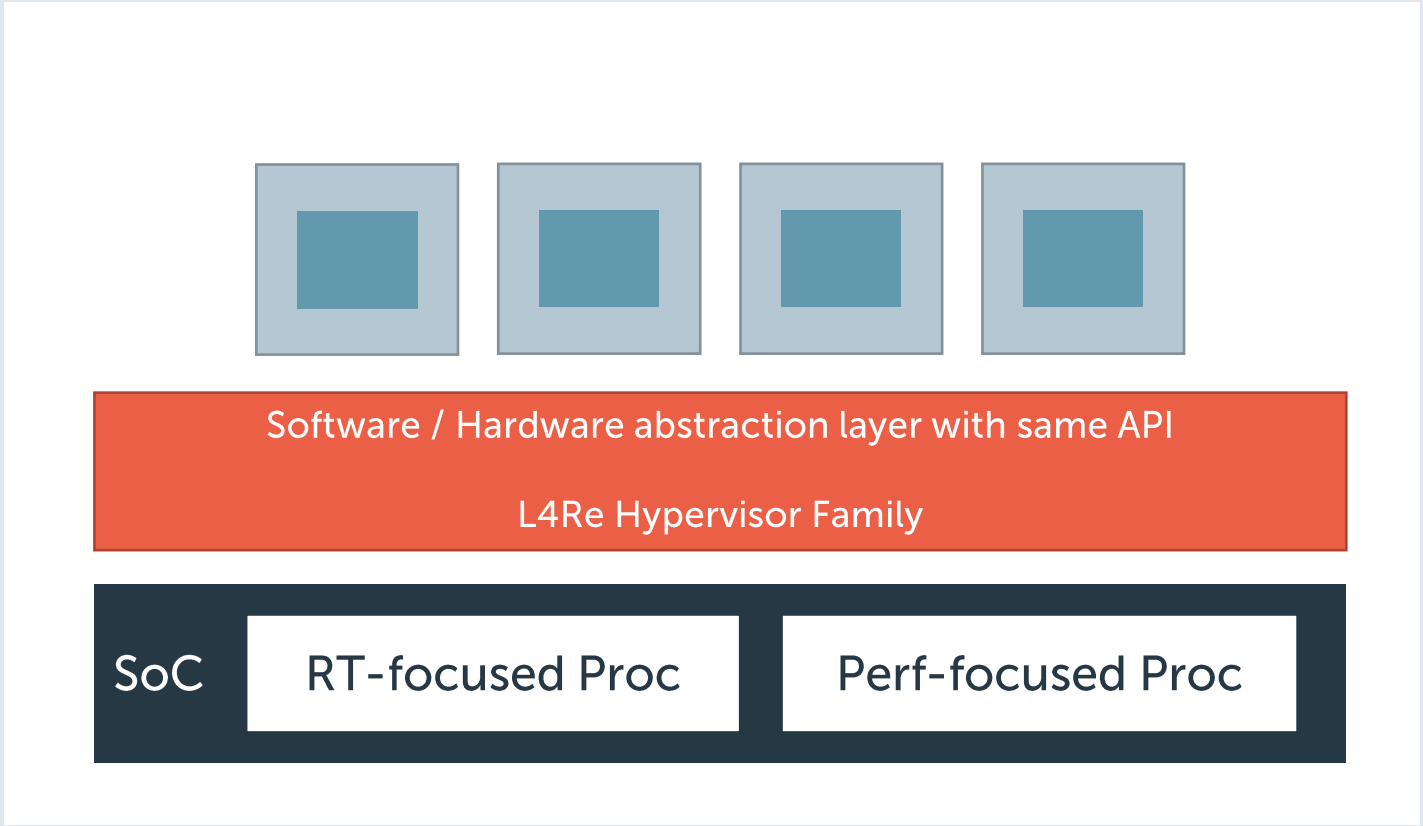
Facilitating SDV

+ **Need for flexible and scalable abstraction layer**
  - For the independence of software functions from the hardware



One Abstraction Layer for all Architectures

+ **Uniform API across all different architectures**
+ **Disconnect application software from specific hardware**
+ **Flexible placement of software components**

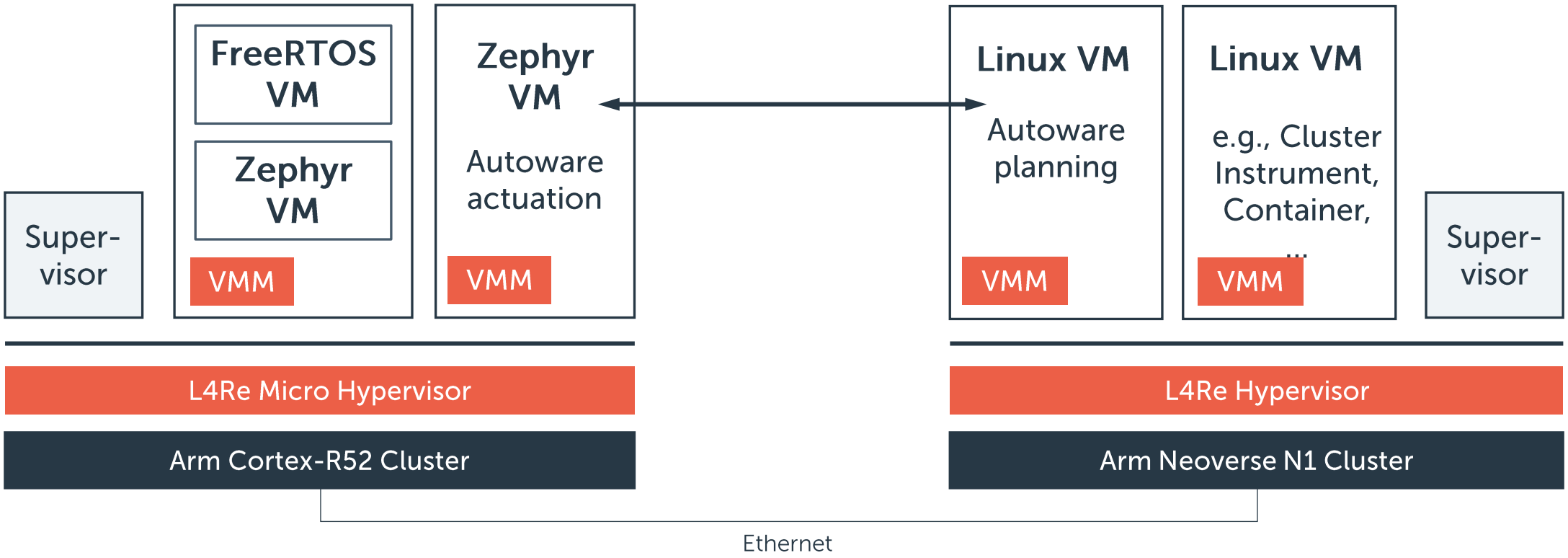


L4Re OPERATING SYSTEM FRAMEWORK

Use-Case: Distributed ADAS

L4Re MICRO HYPERVISOR
on NXP S32Z Cortex-R52 MCU

L4Re HYPERVISOR
on ADLINK AVA Platform



L4Re Operating System Framework

Benefits

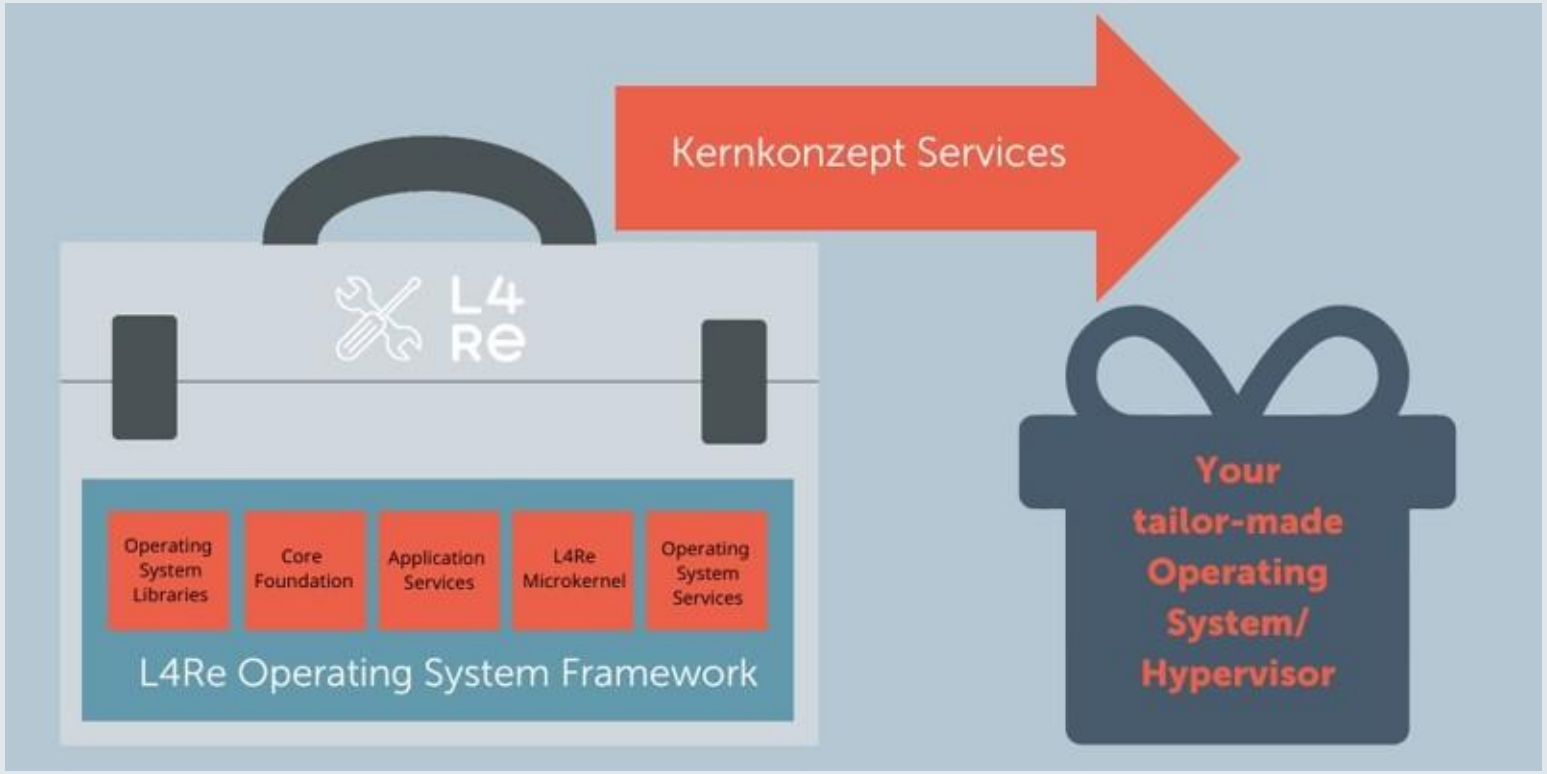
- Flexible and scalable system architecture
- System Security
- Safety
- Available, transparency, trustworthiness
- Strong & manageable path to security & safety certification

L4Re Facts

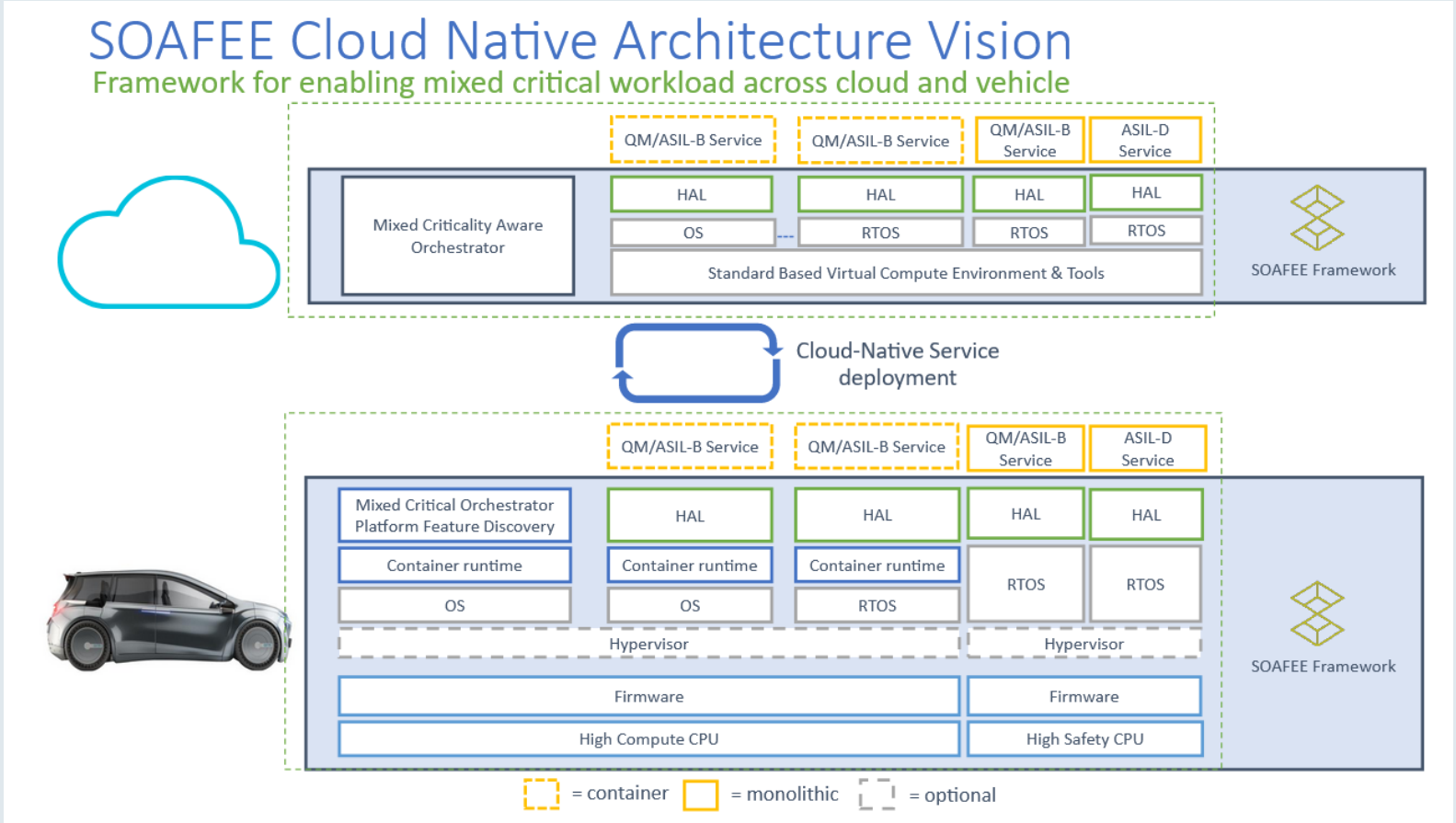
- L4Re Hypervisor Family (L4Re Hypervisor and L4Re Micro Hypervisor)
- Uniform API for L4Re Hypervisor and L4Re Micro Hypervisor
- System security by object-capability-based design
- Small application-specific trusted computing bases (TCBs)
- Modular structure
- Open source

The L4Re Operating System Framework

- The L4Re Hypervisor family is part of the L4Re Operating System Framework



SOAFEE – Shaping the Software Defined Vehicle



https://soafee.io

L4Re Hypervisor Family – Uniform Abstraction

+ **Scalable and flexible abstraction layer**

+ **The L4Re Hypervisor Family is currently available as open source, for:**
  - MPU processors: Cortex-R52: NXP S32Z, ST Stellar, Renesas R-Car, Arm VFP
  - MMU processors: Arm, RISC-V, x86

+ **Uniform API across all architectures**
  - Native L4Re applications in C, C++, and Rust
  - Leverage hardware virtualization to isolate complex and diverse software stacks

+ **Outlook: Common container format from small and static to big systems**

Summary

+ **Uniform abstraction layer: the cornerstone for building the software defined vehicle (SDV)**

+ **Hardware platforms are heterogeneous to cover a wide range of requirements**

+ **Software and hardware must be decoupled for independence**
  - Flexibility, updates, and maintainability to automotive development

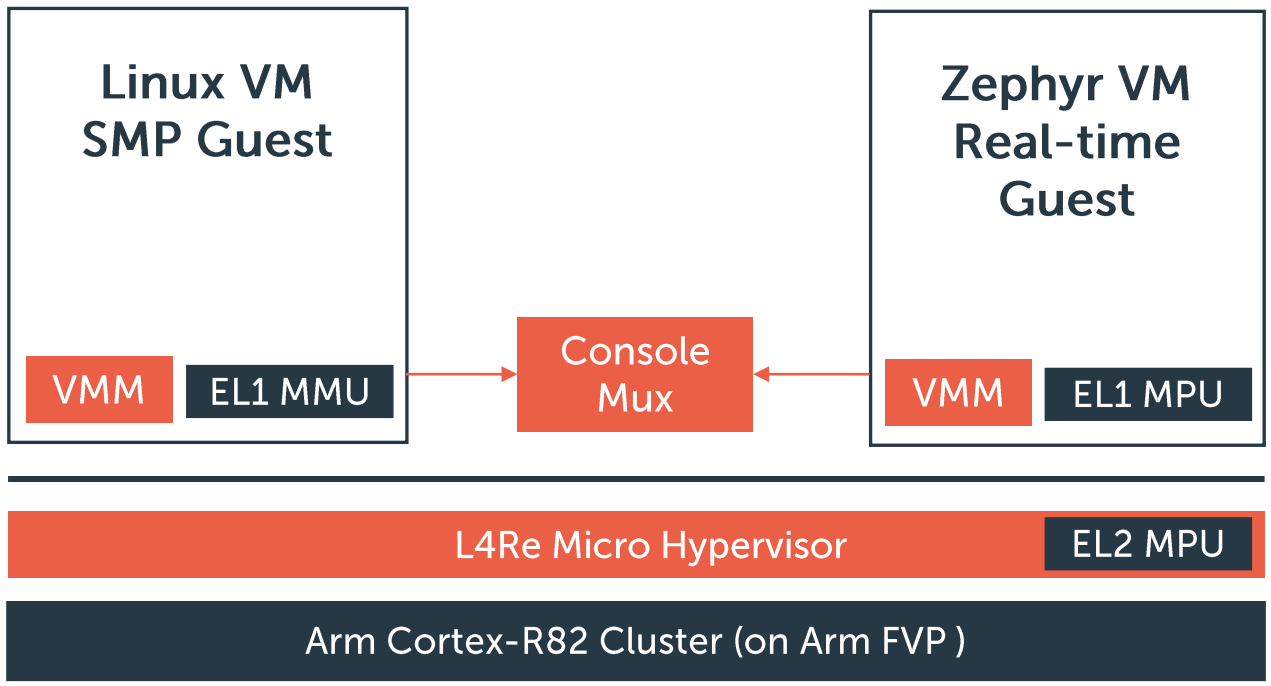
+ **OEMs can reach the scalability and flexibility for the ECU architecture and ECU development with the L4Re Hypervisor Family that provides uniform APIs across the spectrum.**

Q&A

DEMO:

L4RE ON CORTEX-R82, RUNNING LINUX AND ZEPHYR ON L4RE MICRO HYPERVISOR

L4Re Cortex-R82 Demo Setup



+ **Running Linux on Arm Cortex-R**

+ **The upcoming Arm Cortex-R82 processor blurs the line between Cortex-A and Cortex-R processors**
  - Runs Rich-OS, e.g., Linux, on Cortex-R-based hypervisor

Dr.-Ing. Adam Lackorzynski, CTO
adam.lackorzynski@kernkonzept.com

kernkonzept.com / l4re.org