

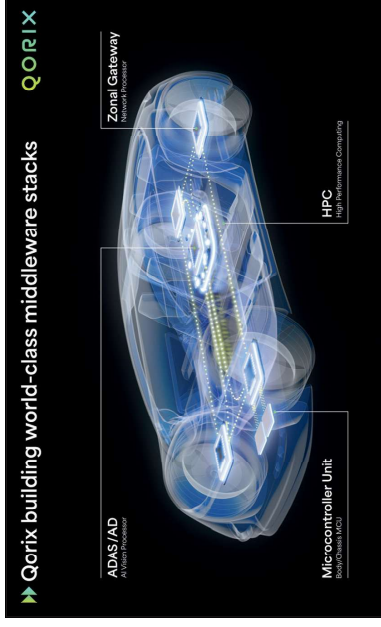


Figure 1: Middleware as an invisible layer in the vehicle. It connects sensors, ECUs, and applications, forming a deterministic overall system. © Qorix GmbH

These four building blocks are essential for every ADAS system. Together, they make safety-critical functions traceable, testable, and auditable — and thus suitable for mass production.

Open Basis for the Industry: What Eclipse S-CORE Aims to Achieve

Eclipse S-CORE (Safe Open Vehicle Core) is an open-source project of the Software-Defined Vehicle Working Group of the Eclipse Foundation. The project's goal is to provide a common, safety-oriented middleware platform for high-performance ECUs. It is developed in close collaboration with OEMs, Tier 1 suppliers, and technology partners (Figure 2).

The basic idea is simple: avoid duplication of work, accelerate development, and establish standards. Rather than each company developing similar basic software functions independently, S-CORE pools industry resources to provide a robust, uniform software core. Manufacturers can then build their own differentiating applications faster and more consistently, with clear links to safety and security (Figure 3).

The focus is on generic services such as interprocess communication, orchestration, logging/tracing, and persistence. Vehicle functions them-

© Qorix GmbH

Eclipse S-CORE

Without Deterministic Middleware, Capabilities of ADAS Are Limited

Reliability and cyber resilience are essential for modern driver assistance systems. Middleware plays a crucial role in this regard. Eclipse S-CORE is an open platform that Qorix is developing into a marketable product on real hardware.

Lars Bauhofer

Although sensors detect the environment and AI models make decisions that are monitored by safety monitors, driver assistance systems quickly reach their limits without a robust execution environment. Data streams must be processed in parallel, functions with different levels of criticality must be isolated from each other, and time specifications must be adhered to strictly and reliably. In addition, there are regulatory requirements for functional safety (ISO 26262) and cybersecurity (ISO/SAE 21434).

One often-overlooked component in this chain is the invisible foundation between the operating system and applications: middleware that enables deterministic execution, reliably

secures communication, and ensures traceable behavior in the event of a fault. This is exactly where Eclipse S-CORE comes in. As part of this open-source initiative, Qorix is helping to develop the core of an automotive software stack that is usable for the industry, along with many other companies (Figure 1).

Middleware as the Foundation: Requirements From an ADAS Perspective

For driver assistance systems to function reliably, the middleware must fulfill four core tasks. First, deterministic runtime orchestration ensures predictable scheduling and resource allocation without blocking time-crit-

Customer Functions

Easing feature integration: Enables shipping new customer functions & apps over-the-air.



Software Platform

Separation of concerns via layered architecture & APIs.



Hardware Platform

Decoupling SW platform from hardware: Enables exchanging hardware whenever necessary.



Figure 2: S-CORE covers all software levels between the SoC's board support package and the application level.

© Eclipse S-CORE Project, Eclipse Foundation



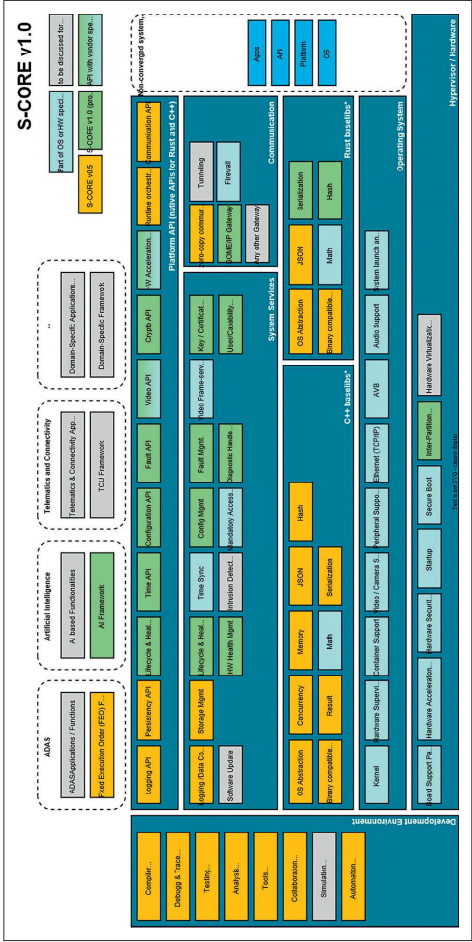


Figure 3: Eclipse S-CORE architecture overview: Generic middleware services form the inner layer between hardware/OS and application software.

© Eclipse S-CORE Project, Eclipse Foundation

and is supplemented with safety documentation, integration support, and long-term maintenance. This transforms collaboratively developed code into production-ready middleware that meets the automotive industry's high standards without losing S-CORE's open character.

Thus, Qorix combines community development with industrial use, offering an open and reproducible solution that is also practical for safety-critical ADAS programs.

S-CORE on Real Hardware: A Demonstrator as Proof of Concept

The Qorix Performance Stack, based on Eclipse S-CORE and developed for use on high-performance ECUs, served as the basis for the demonstrator. It transfers deterministic orchestration, secure interprocess communication, logging/tracing, and persistence from the open-source base to a near-series middleware environment optimized for safety-critical applications, such as advanced driver-assistance systems (ADAS).

Using this technology, Qorix presented the first publicly displayed S-CORE-based performance demonstrator at ELIV 2025 in Bonn in October. The demonstrator showcased the central middleware services under real conditions and demonstrated how open source can be applied to

ADAS programs. This demonstrated the viability of the open-source approach — an important step in testing its transferability to safety-critical ADAS domains and gaining valuable integration experience, such as interaction with hypervisor guests or safety islands.

Open Source and Distribution: Why the Two Belong Together

Open development increases speed, transparency, and reuse, particularly in non-differentiating areas of vehicle software. However, this alone is not enough for the automotive industry. OEMs and Tier 1 suppliers also expect binding processes, such as reproducible builds, long-term maintenance across vehicle generations, and robust artifacts for functional safety and cybersecurity.

This is where the distributor model comes in. It transforms collaboratively developed open-source code into a product ready for mass production with tested and validated builds, continuous security updates, documented evidence of safety and compliance requirements, and adaptable middleware for specific platforms or performance needs.

With the Qorix Performance Stack, S-CORE fulfills this role. The open core of S-CORE is supplemented by building blocks crucial for industrial use, in-

cluding safety documentation and long-term maintenance. This creates a sustainable path for the industry. The community code remains open and modern while providing the stability and industrialization indispensable for series production — without lock-in and with minimal distance to upstream.

What Does “Determinism” Mean in Practice?

In deterministic orchestration, program design (the definition of tasks and data dependencies) is clearly separated from deployment, where allocation, synchronization, and resource utilization are determined. This separation makes processes traceable. Which function runs when, for what reason, and with what guarantees?

Combined with well-defined inter-process communication and systematic logging and tracing, this creates a middleware stack that can be measured, profiled, and audited. This is a basic requirement for building robust safety arguments in ADAS systems and for meeting regulatory requirements.

Equally important is the runtime's ability to handle different levels of criticality in parallel. Safety-critical functions must be prioritized reliably, even

when comfort- or data-intensive services are running simultaneously. For heterogeneous platforms with different operating systems or hypervisor layouts, clear abstraction layers and defined interfaces are required. The “inner layers” of S-CORE play precisely this role.

Conclusion and Outlook

The next generation of driver assistance systems will not be defined solely by better sensors or more powerful algorithms. Middleware is the decisive factor because it makes safety-critical functions deterministic and traceable under real conditions, thus making them reliable for safety arguments.

Eclipse S-CORE is the first open platform to meet this requirement, supported by OEMs, suppliers, and technology partners. Qorix plays a dual role in the project: as a contributor developing central modules such as persistence and orchestration and as a distributor qualifying the open-source core for series production. The first S-CORE-based performance demonstrator shows that this approach is practical today.

The next steps are scaling and validation: providing broader hardware support, ensuring in-depth safety and security, and maintaining long-term support across vehicle generations. If successful, open middleware could become a key foundation for advanced driver-assistance systems (ADAS) and the software-defined vehicle — industry-ready, reproducible, and ready for series production. (AH)

www.qorix.ai

This article is an English translation of a feature originally published in the German magazine Future Mobility (HANSER automotive).



Video: Qorix performance stack on high-performance ECU with mixed-criticality workloads, orchestration, communication modules, and ACC algorithm in a simulation.

© Qorix GmbH



Lars Bauhofer is Product Program Manager at Qorix GmbH.

© Qorix

Sie wollen alle News aus der Automobilelektronik?

Ihre Vorteile auf einen Blick:

- ☞ Thema der Woche, aktuelle Branchennews, Trends
- ☞ Das Beste aus den Fachmagazinen HANSERautomotive, OEMSupplier & Mobile Automation
- ☞ Interviews, Termine, Produktneuheiten und viel mehr



Gleich kostenlos anmelden:
www.hanser-automotive.de/newsletter

