Pioneering efficient automotive mixed-criticality software design

Red Hat and Vector are collaborating to create an example blueprint on how mixed-criticality workloads can be efficiently achieved on a single-host, Linux-based operating system.

The blueprint uses Red Hat In-Vehicle OS to allow mixed-criticality management and integrates Vector's MICROSAR Adaptive safety automotive middleware, which provides functionality such as communication, diagnostics, and security. This blueprint will be implemented on Qualcomm's SA8775P hardware, which serves as a high-performance computing platform.

Red Hat® In-Vehicle Operating System (OS) demonstrates its ability to manage mixed-criticality workloads by partitioning a single OS into 2 software-based partitions: 1 for safety workloads and 1 for quality management (QM). MICROSAR Adaptive runs in the safety space outside the QM partition, supporting safety-critical and hardware-independent applications for high-performance platforms. This integrated approach lets automotive original equipment manufacturers (OEMs) streamline software architectures, optimize resource use, and simplify the deployment of AUTOSAR Adaptive-based safety applications. Additionally, solutions like Red Hat OpenShift® enhance data processing and visualization, showcasing the flexibility of Red Hat and Vector's broader portfolio in supporting next-generation, cloud-native automotive architectures.

Industry challenge

As automotive systems evolve toward high-performance computing platforms, manufacturers face challenges in managing mixed workloads, particularly with safety-critical and nonsafety applications running on shared hardware. Historically, virtual machines (VMs) have been widely used to address these challenges, providing strong isolation and support of existing workloads. However, relying exclusively on VMs can lead to:

- Increased complexity in system management.
- Limited flexibility for scaling applications.
- Dependence on hardware isolation reduces adaptability.

Red Hat In-Vehicle Operating System complements the use of VMs by offering an alternative approach for certain scenarios. It provides efficient management of mixed-criticality applications using containers. Containers provide robust isolation, lightweight deployment, and dynamic resource management, making them ideal for separating safety-critical and nonsafety workloads. At the same time, VMs remain an option for cases requiring strong hardware-based isolation, the consolidation of legacy systems, or platform-specific management functions.

This flexibility allows automotive OEMs to:

- Run mixed-criticality applications in containers, ensuring freedom from interference (FFI) while optimizing resource usage and scalability.
- Use containers for lightweight, efficient deployment of modern safety and nonsafety applications.
- Use VMs for hardware-based isolation or to support non-Linux® or existing systems where required.
- Design architectures that balance efficiency, scalability, and compliance with safety and security standards.

Achieving mixed criticality without VMs

Red Hat In-Vehicle OS lets mixed-criticality workloads safely coexist on a single platform through:

1. **Partitioned architecture:** The system uses software-defined partitions to separate safety-critical (ASIL B) and nonsafety (QM) workloads on the same hardware. This logical separation ensures that nonsafety applications cannot interfere with safety-critical functions, supporting the requirements of the functional safety standard ISO 26262.
2. **Container-based isolation:** As an alternative to VM-based separation, Red Hat In-Vehicle OS uses containers to isolate mixed-criticality workloads. Containers rely on kernel-native technologies such as namespaces and control groups (cgroups) to enforce isolation between safety and nonsafety workloads while minimizing resource overhead and simplifying lifecycle management.
3. **Controlled resource allocation and FFI:** Red Hat In-Vehicle OS uses cgroups to enforce central processing unit (CPU) and memory limits, ensuring that safety-critical tasks retain access to the resources they need, even under high load. Combined with namespace-based isolation, this ensures that nonsafety workloads cannot interfere with safety-critical operations. This predictable system behavior is essential for maintaining FFI.

Preintegrated blueprint: Vector MICROSAR Adaptive and Red Hat In-Vehicle OS

The goal of this blueprint is to provide an example of how 2 products can be effectively combined, as a starting point for new projects, reducing development and integration efforts for OEMs.

1. **Red Hat In-Vehicle OS for Mixed-Criticality Workloads**
   - Red Hat In-Vehicle OS is a production-grade, safety-focused, open source Linux platform purpose-built for SDVs. Built on strong foundations of industry-leading Red Hat Enterprise Linux, it combines safety with automotive-grade optimizations.
   - Red Hat In-Vehicle OS has been certified to meet selected requirements of ISO 26262:2018 Parts 3 and 6 for mixed-criticality support. This certification demonstrates Red Hat's ability to support the coexistence of safety-critical applications (up to ASIL B) and nonsafety applications on a unified kernel and shared hardware, with appropriate isolation mechanisms in place.
   - FFI capabilities of Red Hat In-Vehicle OS are demonstrated in this example blueprint through the coexistence of mixed-criticality workloads aligned with industry standards.
2. **Advantages of Vector's MICROSAR Adaptive**
   - Middleware for mission-critical applications: Supports safety-critical applications up to ASIL D, making it an ideal choice for advanced use cases such as Autonomous Driving (AD) and Advanced Driver Assistance Systems (ADAS).
   - Automotive network connectivity: MICROSAR Adaptive allows the integration of HPCs into existing automotive networks and infrastructure by its unique support of automotive protocols. The protocol toolbox includes all major automotive protocols, such as SOME/IP with Service Discovery, logging, UDS, and SOVD, but also high-precision time synchronization or calibration.

- Efficient application reuse and collaboration: MICROSAR Adaptive allows OEMs to maximize efficiency by reusing applications across projects while fostering collaboration on a global scale. Built upon essential automotive standards, MICROSAR Adaptive offers a reliable and versatile platform for high-performance ECUs.
- High-performance Communication: Diverse applications across the entire vehicle network are connected, ensuring efficient communication through high-performance zero copy IPC for internal processes.
- Central software link: MICROSAR Adaptive provides integration within existing embedded and automotive-grade software ecosystems. It supports well-established systems like AUTOSAR Classic and Hypervisor solutions, while also accommodating emerging applications in advanced driver assistance systems (ADAS), multimedia, and image processing.
- Regular software updates: Orchestration of updates of the diverse software running on today's HPCs.

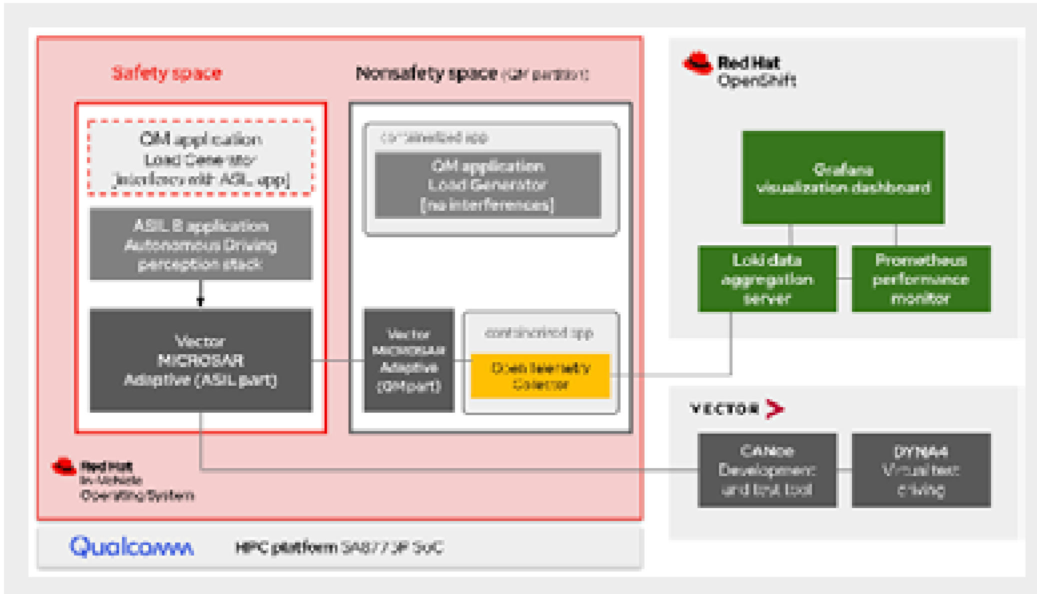
3. **Implemented on the Qualcomm SA8775P platform**
   - The Qualcomm SA8775P platform serves as a high-performance computing environment for this blueprint, providing the representative hardware needed to showcase Red Hat In-Vehicle OS and MICROSAR Adaptive capabilities.

Key Benefits

1. **Efficient resource management**
   - Red Hat In-Vehicle OS ensures optimized performance by reducing dependency on resource-heavy VMs, simplifying architecture, and increasing system agility.
2. **Accelerated deployment of safety-critical features**
   - The pre-integrated blueprint minimizes time-to-market, reducing the effort required for integration.
3. **Robust mixed-criticality isolation**
   - By using software-based isolation, Red Hat In-Vehicle OS ensures FFI, crucial for automotive safety applications.

Demonstration Overview

The blueprint will highlight how Vector's MICROSAR Adaptive Safe middleware and Red Hat In-Vehicle OS work together to manage safety-critical and nonsafety workloads on the Qualcomm automotive platform.



Safety Workload:

- The blueprint includes a sensor application that shall be used as a placeholder for autonomous driving (AD) and ADAS applications. These applications run on Vector's MICROSAR Adaptive Safe automotive middleware, which provides functionality such as communication, diagnostics, and security.
- The applications compute sensor data based on the virtual vehicle's movement in a simulated driving environment, which is generated by Vector's DYNA4. It runs offboard and outside of Qualcomm's SA8775P board.
- Vector's development and test tool CANoe for hardware-in-the-loop (HIL) and software-in-the-loop (SIL) projects, in combination with Vector's DYNA4 simulation tool for virtual test driving, complete the demonstration environment.

Nonsafety workload: The automotive platform also runs nonsafety workloads alongside the safety-critical AD/ADAS applications. These workloads include:

- A load generator that applies additional processing demands on the system. When the load generator is configured as a safety workload, it affects the performance of the AD/ADAS application, demonstrating interference in a shared safety-critical environment. In contrast, when the same load generator runs in the QM partition, it does not affect the AD/ADAS application, showcasing the robust isolation provided by the system.
- A telemetry stack that monitors the performance of safety-critical workloads. In this demonstration, the telemetry collector operates outside of the QM partition, running in the ASIL partition to ensure monitoring of critical system metrics.

Performance monitoring: The telemetry data collected on the platform is sent to an external machine for processing and visualization. On the off-board side, using Red Hat OpenShift, the open observability platform visualizes real-time metrics, confirming that safety-critical workloads are unaffected by concurrent nonsafety processes, even under load. This highlights the ability for Red Hat In-Vehicle OS to enforce robust isolation and prioritize critical workloads, while demonstrating how off-board tools can complement the platform for comprehensive data analysis.

Summary

Vector and Red Hat showcase how Red Hat In-Vehicle OS facilitates mixed-criticality workloads, running MICROSAR Adaptive safety-critical applications alongside nonsafety workloads. This collaboration underscores the scalability of Red Hat In-Vehicle OS for high-performance platforms and its integration with proven automotive middleware. By using Red Hat In-Vehicle OS, Vector's MICROSAR Adaptive, and Qualcomm hardware, automotive manufacturers can optimize safety, scalability, and efficiency for next-generation vehicles.

For details and a demonstration

- Learn more about Red Hat's Automotive offerings
- Learn more about Vector's Automotive offerings

facebook.com/redhatinc
@RedHat
linkedin.com/company/red-hat

North America
1 888 REDHAT1
www.redhat.com

Europe, Middle East, and Africa
00800 7334 2835
europe@redhat.com

Asia Pacific
+65 6490 4200
apac@redhat.com

Latin America
+54 11 4329 7300
info-latam@redhat.com