Building Performance ECUs with Adaptive AUTOSAR
## Major market trends and their impact

<table>
<thead>
<tr>
<th>Trends</th>
<th>Impact on E/E architecture</th>
<th>Impact on SW architecture</th>
<th>Industry need</th>
</tr>
</thead>
</table>
| E-Mobility      | Shrink of powertrain reduces hw complexity | • Software complexity increase  
• Central car computer approach | Adaptive AUTOSAR is the base technology for  
• safe  
• secure  
• flexible  
• up-to-date  
• high performance in-car computers. |
| Automated Driving | High data volumes  
Safety meets performance | • Safety on high performance ECUs  
• Fail operational systems | |
| Mobility Service | Car-to-X connectivity  
Update over the air | • Holistic security approach  
• Service-oriented architecture (SOA)  
• Unsupervised partial updates | |
Requirements for performance ECUs

Software over the air updates
- New vehicles features
- Updates and patches
- Silent testing

Dynamic deployment
- Migration from „classic ECU“ to high performance controller (HPC) or backend
- Network accessible sensors & actuators

Remote analytics and diagnostics
- Remote diagnostics
- Predictive diagnostic
- Fleet campaigns

Developer oriented, target independent environment
- Environment independent software
- Easy qualification and deployment
- Small, encapsulated and exchangeable software services (microservice)

Dependable systems
- Safety
- Security
- Availability
- Reliability
- Maintainability
Consolidated vehicle infrastructure architecture
Use-case remote update

Architectural principles:
• Central external connection
• Distribution of updates across multiple ECUs

Supporting features:
• Coordinated A/B Update across ECUs
• Secure communication
• Application containerization
• Layered security architecture
Use-case ADAS

Architectural principles:
- Separation between planning and mechatronic parts
- Hierarchical safety architecture

Supporting features:
- ASIL-B performance platform
- ASIL-D classic platform
- Hierarchical runtime supervision
Adaptive AUTOSAR alone is not the solution.

Neither is Classic.
High-performance computer – Software architecture

Complex software system on heterogeneous performance controllers
- Update scheme for applications, OS instances, hypervisor
- Distributed health management
- ECU state management spanning Adaptive and Classic instances
- ...

- Secure startup, authentication
- Safety-relevant vehicle functions, monitoring of performance partitions

- Takeover of existing vehicle functions from Classic AUTOSAR
- Secure startup, authentication
- Safety-relevant vehicle functions, monitoring of performance partitions

- Novel user functions: e.g. App Store
- New CPU-intensive (safety-relevant) functions: e.g. sensor fusion
- Complex software system on heterogeneous performance controllers
  - Update scheme for applications, OS instances, hypervisor
  - Distributed health management
  - ECU state management spanning Adaptive and Classic instances
  - ...

- POSIX OS
- POSIX OS
- Virtual Machine
- Virtual Machine
- Virtual Machine
- Virtual Machine
- Hypervisor
- Performance Partitions
- Security Partition
- Safety Partition
- Secure Boot
- Performance Cores
- Safety Cores
Example: Distributed updates

- Vehicle Functions Partition:
  - Container
  - Diagnostic Client
  - Update Service App.
  - Software Configuration Manager
  - Execution Manager
  - Adaptive AUTOSAR on Linux

- Privileged Partition:
  - Container
  - Vehicle Function
  - Diagnostic Manager
  - Adaptive AUTOSAR on Linux

- Classic AUTOSAR Components:
  - Dcm
  - Classic AUTOSAR
  - Lockstep Safety OS

- Hypervisor
- Bootloader

- Diagram notes:
  - Transfer Images/Pass control
  - Flashing of software

- Example: Distributed updates
  - Example: Distributed updates
Example: Distributed health management
More aspects with relevant interaction

**ECU state management**

- Coordinated shutdown/sleep
  - Between multiple Adaptive and Classic instances
  - Use of OS sleep states
  - Hypervisor
- Network management
  - With potentially virtualized Ethernet devices

**Logging, tracing, debugging and testing**

- Relating multiple logs across ECUs for debugging complete event chains
- Compatibility of test tools across different stacks
- Synchronized debugging

**Security Architecture**

- Secure boot
- Secure partial updates
- Secure communication
- Mostly involve dependencies between hardware, operating system, and AUTOSAR stack
- Must be compatible on the network
AUTOSAR as foundation for software systems

- Classic and Adaptive AUTOSAR form a foundation for complex automotive software systems

- System functionality must be established across individual AUTOSAR instances

- System properties must be ensured through system architecture, particularly for Safety // Security // Reliability

- Software platform must be maintained beyond deployment of the vehicle
High-performance computer – One-stop solution
Conclusion

• Adaptive AUTOSAR is not the solution for all performance controllers

• Performance controllers build on a software system architecture on the basis of AUTOSAR

• Use of proven reference architecture reduces risk in system design and integration

• Tooling and development environment are key to master complexity
Thank you for your attention!

Dr. Moritz Neukirchner

moritz.neukirchner@elektrobit.com