

# Paving the Way for the “Software Defined Vehicle”

- A fundamental paradigm change towards IoT mobility

**Martin Schleicher** <sup>1)</sup>

*1) Continental AG*

*Guerickestrasse 7, 60488 Frankfurt, Germany, (E-mail: martin.schleicher@continental.com)*

**KEY WORDS: software defined vehicle, cloud-native, collaboration, vehicle architecture**

The automotive mega trends "CASE" (Connected, Automated, Shared, Electrification) have one thing in common: they are dependent on electronics and even more on software. The "Software Defined Vehicle" enables new consumer functions and user experiences through software. The increased importance of software has a significant impact for the automotive industry.

EE architectures are going to evolve in several steps from current distributed systems to domain-centralized networks with a central HPC per domain to zonal (zone-based) architectures.

To successfully master the "software defined vehicle", the automotive industry needs to adopt thinking and acting from software companies and master several dimensions of software: architecture, development methods, culture, and business models.

The main targets car manufacturers and suppliers want to achieve with the Software Defined Vehicle are:

- Enable Digital Lifecycle business models such as “Software as a Product”, “Software as a Service” as well as “Function on Demand”
- Software separation from hardware
- Shorter time to market and product development cycles
- Shorten software development, and integration cycles
- Virtual development and testing
- Data-driven development

To support these goals, on the one hand the system architecture can be described as a 3 layer architecture with: Hardware, Operating System/Middleware and Application or Function layer. On the other hand, it is also important how the software is being built and deployed to a vehicle:

- Service Oriented Architecture enabling SW-HW separation
- Containerization to support isolation and density of software
- Cloud enablement
- (Over-the-Air) Updates for continuous software deployment to vehicles
- Agile and Lean Methods used in development
- DevOps, Continuous Integration and Automated Testing

With the "Continental Automotive Edge Framework" Continental is creating the foundation of the transformation towards the software defined vehicle, consisting of the following building blocks:

- Scalable high-performance computer hardware
- Vehicle operating system and Middleware
- Cloud software
- Toolchain and DevOps Workbench

The Software Defined Vehicle is a fundamental paradigm change towards IoT mobility. Actually it can be seen as third revolution for the development of vehicles after the introduction of the assembly line by Ford and Toyota introducing Lean Production. The SDV requires

- A modular framework architectures to optimize re-use, time-to-market and cost.
  - Definition and management of interfaces is key for speed and maturity of SW development.
  - Cloud is essential, cloud-native concepts (technology, processes) will become standard also for in-vehicle system.
  - Cross industry collaboration in ecosystems will bring the SDV successfully to the road.



Software Defined Vehicle