Vehicle OS

Jun, Zhang
AUTOSAR China Day

16th Mar 2023  Shang Hai
Towards a Centralized/Zonal Architecture

**Function Architecture**

- Reduce cost
  - ECU consolidation
  - Simplified wiring
- Reduce CO₂ footprint
  - Lighter wiring harness
- Enable the software defined vehicle
  - Decouple hardware from software
  - New (purely) software-driven vehicle functions
  - Increase the value of a vehicle over its lifetime
Software Defined Vehicle

In context of the E/E system, there are three enablers for the software defined vehicle:

- **Architecture:** HPCs and zonal ECUs
- **Hardware:** High-performance microcontrollers and microprocessors
- **Software:** Powerful software platform and ecosystem → **Vehicle OS**
  - To cope with the increasing SW complexity, mainly in HPCs and zonal ECUs, controlled by OEMs
  - Separate software solution for small ECUs, potentially developed out of context, controlled by Tier1s
A Vehicle OS is a development and operations platform for services and applications of all vehicle domains. It consists of a Base Layer and a Software Factory and supports collaboration between companies.

- The Vehicle OS runtime software is called Base Layer and its instantiation may differ from target to target (e.g., microcontroller, microprocessor, and backend).

- As Vehicle OS infrastructure, the Software Factory supports and automates the developer’s journey to develop, integrate and deploy Base Layer and applications.

- Close and agile Collaboration (C) between OEM and suppliers via a supporting platform is key for success.
Software Factory

Middleware & System Functions

Infrastructure & OS

Vehicle OS | 16-Mar-23 | 7
Software Factory: Workflow and Tools

Application Development & Test Workflow
- App. Dev. Tools
- App. Test Tools

System Design Workflow
- System Design Tools

Application Provision

ECU Integration & Test Workflow
- ECU Int. Tools
- ECU Test Tools

Backend Integration & Test Workflow
- Backend Int. Tools
- Backend Test Tools

Vehicle Integration & Test Workflow
- Vehicle Integration Tools
- Vehicle Test Tools

Software Deployment and Data Management Workflow
- Software Deployment and Data Management Tools

Base Layer Provision potentially pre-integrated

Software Factory

Vector contribution

3rd-party contribution
The Vehicle OS Is the Next Step in Simplifying Our Customer Life

For our customers, we see big benefits when following the Vehicle OS approach

- **Base Layer**
  - With a Base Layer, Vector provides aligned building blocks for a safe and secure embedded runtime software
  - This goes beyond single products like MICROHAR Classic or MICROHAR Adaptive
  - If desired, a Base Layer can also be pre-integrated by Vector (e.g., in software platform scenarios)
  - This includes the integration with 3rd party products like POSIX OS or microprocessor hypervisor

- **Software Factory**
  - Big ECUs like HPCs run hundreds of applications, developed asynchronously all over the world
  - Additionally, today’s software developers want to follow a feature-based development approach by using e.g., well-known Git workflows
  - Without a high degree of automation, software integration and test become major pain points in this scenario
  - Our answer to this challenge is the Software Factory enabling a highly automated software integration and test process according to the DevOps principle
  - The result is a scalable development environment that enables focusing on application development

**Vector’s strategy**

- For Vector, the Vehicle OS is a common vision shared between many products of different product lines, and not limited to embedded software
- Many products are and will be aligned to this vision with the goal of providing a powerful software platform and a corresponding development and operations ecosystem
Collaboration & Summary

Let’s shape the future of automotive software together!
Vector Vehicle OS Symposium 2022

Friday, October 28, 2022

12:00 p.m. CEST
Marcelino Vargas | Vector
- Model based E/E system design
- Contribution of the system design to the vehicle OS concept
- PREEvision as system design tool

12:30 p.m. CEST
Lunch Break & Exhibition
Time for networking & visiting the exhibition

1:30 CEST
Cloud-Native Technologies for In-Vehicle Software
Andreas Rasch | Vector
- Applying cloud native paradigms to automotive software
- Deployment in the heart of the workflow
- Where are we today - what’s next?

3:15 p.m. CEST
Coffee Break - Exhibition - Get Together with Open End
Time for networking & visiting the exhibition
For more information about Vector
And our products please visit

www.vector.com

Jun, Zhang
Vector China