AUTOSAR Introduction
The vision, the partnership and current features in a nutshell

Presenter

Occasion (Meeting, Conference, etc.)

27 October 2020

Location @ Host
Agenda

- Introduction of the AUTOSAR Partnership
- Challenges in the automotive industry
- Architecture and recent features
- Smart solutions based on AUTOSAR
- Processes and quality
- Outlook
Agenda

- Introduction of the AUTOSAR Partnership
- Challenges in the automotive industry
- Architecture and recent features
- Smart solutions based on AUTOSAR
- Processes and quality
- Outlook
AUTOSAR Introduction

is a worldwide development partnership of car manufacturers, suppliers and other companies from the electronics, semiconductor and software industry.
AUTOSAR aims to improve complexity management of integrated E/E architectures through increased reuse and exchangeability of SW modules between OEMs and suppliers.

**Exchangeability between manufacturers’ applications**
- **Supplier A**
  - Chassis
  - Safety
  - Body/Comfort
- **Supplier B**
  - Chassis
  - Safety
  - Telematics
- **Supplier C**
  - Body/Comfort
  - Powertrain
  - Telematics

**Exchangeability between suppliers’ solutions**
- **Platform a.1, a.2, a.n**
- **Platform b.1, b.2, b.n**
- **Platform c.1, c.2, c.n**

**Exchangeability between vehicle platforms**
- **Platform e.1, e.2, e.n**
- **Platform d.1, d.2, d.n**
Aims and benefits of using AUTOSAR

AUTOSAR aims to standardize the software architecture of Electronic Control Units (ECUs). AUTOSAR paves the way for innovative electronic systems that further improve performance, safety and security.

- Hardware and software – widely independent of each other.
- Development can be decoupled (through abstraction) by horizontal layers, reducing development time and costs.
- Reuse of software enhances quality and efficiency.
Exploitation of the standard provides significant benefits

- Establish development distribution among suppliers
- Compete on innovative functions with increased design flexibility
- Simplify software and system integration
- Reduce overall software development costs

An industry standard is established.

OEM

Supplier

Tool provider

New market entrant
Exploitation of the standard provides significant benefits

- Enable more efficient variant handling
- Reuse software modules across OEMs
- Increase efficiency of application development
- Invent new business models

An industry standard is established.
Exploitation of the standard provides significant benefits

An industry standard is established.

• Interface with development processes
• Embed tools into an overall tool environment
Exploitation of the standard provides significant benefits

- Enable new business models by means of standardized interfaces
- Easily understand how automotive software is developed

An industry standard is established.

OEM

Supplier

Tool provider

New market entrant
More Than 280 AUTOSAR Partners

9 Core Partners

BMW Group  |
BOSCH  |
Continental  |
DAIMLER  |
Ford  |
GM  |
PSA  |
TOYOTA  |
VOLKSWAGEN  |

56 Premium Partners

altran  ·  APTIV  ·  arm  ·  Blackberry  ·  Capgemini  ·  Continental  ·  Deloitte  ·  dSPACE  ·
ETAS  ·  Fraunhofer  ·  HCL  ·  HONDA  ·  HYUNDAI  ·  Infineon  ·  INTRON  ·  Intel  ·  itk  ·
JTEKT  ·  KIT  ·  L&T Technology Services  ·  LEAR  ·  MathWorks  ·  NEC  ·  NXP  ·  Panasonic  ·
Renesas  ·  SCSK  ·  sodius  ·

51 Development Partners

ABUP  ·  avelabs  ·  BASELABS  ·  Broadyx  ·  C2A  ·  clarinox  ·  codeplay  ·  Corecon  ·  Esl-/Jacc  ·  EYE  ·
Fretecht  ·  Haneic  ·  Intrepid  ·  Inchron  ·  ipCam  ·  JST  ·  IXIA  ·  KRONOSAFE  ·
LAUTERBACH  ·  MAT RICKZ  ·  NCEs  ·  NORDSYS  ·  opensynergy  ·  PLS  ·  PopcornSAR  ·  PROQA  ·  RTAE  ·
RTi  ·  SAFERIDE  ·  SEMOYE  ·  Siemens  ·  Softing  ·  GYYGO  ·  TTECH  ·

2 Strategic Partners

LG  ·  DENSO  ·

+ 144 Associate Partners
+ 24 Attendees
History and current state – Total: 284
Geographical distribution of partners
(last update 02.09.2020)

North America: 35
- Core: 2
- Strategic: 0
- Premium: 6
- Development: 4
- Associate: 23
- Attendee: 0

Europe: 137
- Core: 6
- Strategic: 0
- Premium: 27
- Development: 29
- Associate: 55
- Attendee: 20

Africa: 2
- Core: 0
- Strategic: 0
- Premium: 0
- Development: 2
- Associate: 0
- Attendee: 0

Asia: 112
- Core: 1
- Strategic: 2
- Premium: 23
- Development: 16
- Associate: 66
- Attendee: 4
AUTOSAR Organization

**Legend**
- Core Partner
- Core and Strategic Partner
- Core, Strategic, Premium and Development Partner

**Official AUTOSAR roles**
- Governance
- Strategic Management
- Technical Steering
- Standardization

**Executive Board**
- Legal Team
- Steering Committee
- Communication Team
- Project Leader Team
- Working Groups

**Open Source CB**
- OSS-CB

**Change Control Board**
- CCB

**Consensus Finding CCB**
- CF-CCB

*27 October 2020*
User Group Structure

- **UG-CN (China)**
  - Demonstrator development
  - BSW integration

- **UG-NA (North America)**
  - Common Training: OEM-Tier1 Workflows/ Security
  - Safety
  - Ethernet

- **UG-IE (Improved Exploitation)**
  - 6 thesis on optimizing the use of AUTOSAR
  - Establishment of Adaptive Platform

*27 October 2020*
<table>
<thead>
<tr>
<th>Group</th>
<th>Type</th>
<th>Initiated by</th>
<th>Main Tasks</th>
<th>Coaching</th>
<th>Contact person</th>
<th>Reporting to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead WG</td>
<td>Permanent</td>
<td>PL Team</td>
<td>- Technical supervision of WGs and concept groups&lt;br&gt;- Harmonization of technical content&lt;br&gt;- Assignment of concepts to WGs</td>
<td>PL Coach</td>
<td>WG Speaker</td>
<td>PL Team</td>
</tr>
<tr>
<td>WG</td>
<td>Permanent</td>
<td>PL Team</td>
<td>- Expert statements&lt;br&gt;- Responsibility for documents and code development&lt;br&gt;- Assessment of concepts</td>
<td>PL Coach</td>
<td>WG Speaker</td>
<td>PL Team</td>
</tr>
<tr>
<td>Subgroup</td>
<td>Permanent</td>
<td>WG</td>
<td>WG substructure for a dedicated branch of WG content</td>
<td>-</td>
<td>Subgroup Speaker</td>
<td>WG</td>
</tr>
<tr>
<td>Control Board</td>
<td>Permanent</td>
<td>PL Team</td>
<td>Controlling task within AUTOSAR processes (e.g. CM, QM, SDE or RM) such as CCB, CF-CCB and OSS-CB</td>
<td>PL Team</td>
<td>CM, QM or RM</td>
<td>CM, QM or RM</td>
</tr>
<tr>
<td>Concept Group</td>
<td>On demand (project-based)</td>
<td>CP/SP/PP/DP</td>
<td>Creation and processing of a concept or a set of concepts according to concept process&lt;br&gt;Responsibility for concept validation</td>
<td>-</td>
<td>Concept Owner(s)</td>
<td>Lead WG, WG, PL Team; cf. concept handling process</td>
</tr>
<tr>
<td>Task Force</td>
<td>On demand (project-based)</td>
<td>WG or PL Team</td>
<td>Dedicated task with defined due date</td>
<td>-</td>
<td>TF Speaker</td>
<td>WG or PL Team</td>
</tr>
</tbody>
</table>
AUTOSAR Deliverables

Legend:
- Released as an own standard
- Released as part of the standard it is extending
- A extends B
- A planned to extend B

Acceptance Test

Application Interfaces

Sensor Interfaces

Classic Platform

Adaptive Platform

Foundation
Agenda

- Introduction of the AUTOSAR Partnership
- Challenges in the automotive industry
- Architecture and recent features
- Smart solutions based on AUTOSAR
- Processes and quality
- Outlook
Challenges – selected main drivers

- Highly automated driving
Challenges – selected main drivers

- Car-2-X applications
- Internet of Things and cloud services
Challenges – selected main drivers

- Increasing data rates
Challenges – selected main drivers

- New processor technologies
Challenges – selected main drivers

» Trust
Challenges – Driving changes in E/E Architectures

New types of in vehicle computers are required to fulfill the needs of
• performance,
• flexibility and
• connectivity

But
• backwards compatibility with existing solutions,
• fulfillment of increasing requirements for safety and security
is a must as well.
### AUTOBORG's answer to the upcoming challenges

#### AUTOBORG Classic Platform
- **Real time Requirements**: High, in the range of micro-sec
- **Safety Criticality**: High, up to ASIL-D
- **Computing power**: Low, ~1000 DMIPs

#### AUTOBORG Adaptive Platform
- **Real time Requirements**: Mid, in the range of milli-sec
- **Safety Criticality**: High, at least ASIL-B
- **Computing power**: High, > 20,000 DMIPs

#### Infotainment
- **Real time Requirements**: Low, in the range of sec
- **Safety Criticality**: Low, QM
- **Computing power**: High, ~10,000 DMIPs
AUTOSAR Adaptive Platform
The 3 Pillars of the Adaptive Platform …

I – Safe & Secure
- external communication
- in-vehicle communication
- Platform
  - SecOC
  - IPsec
  - process separation
  - resource budgeting
  - safe data storage
  - supervision
  - exception-less APIs

II – Connected
- Service-Oriented Communication
- Cloud
  - Adaptive Application
  - SOME/IP
  - ComM
  - REST

III – Dynamic & Updateable
- App Development
  - Software Package
  - Executable
  - Manifest
  - UCM
  - ExM

… are the prerequisite for ADAS applications
# AUTOSAR Classic Platform

The 4 Pillars of the Classic Platform...

<table>
<thead>
<tr>
<th>I – Functional Safety</th>
<th>II – Efficiency</th>
<th>III – Field Proven</th>
<th>IV – Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>▪ Mature safety features (e.g. watchdog, E2E communication protection,…)</td>
<td>▪ AUTOSAR stacks from different vendors</td>
<td>▪ Mature by many years of application</td>
<td>▪ Hard real time capabilities</td>
</tr>
<tr>
<td>▪ Scalable from QM up to ASIL D</td>
<td>▪ Cost effective by supporting a wide range of µControllers</td>
<td>▪ High quality due to widespread implementations</td>
<td>▪ Event triggered applications</td>
</tr>
<tr>
<td></td>
<td>▪ Flexibility due to CDD</td>
<td>▪ Established development processes</td>
<td>▪ Flexible by supporting a wide range of protocols and networks</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ Scalability by configuration</td>
</tr>
</tbody>
</table>

... to form the standard solution for today’s automobiles
Strengthen and Extend Pillars of AUTOSAR

**AUTOSAR USP**

**Strengthen Safety & Security**
- Support standard failure and attack scenarios
- Extend test and verification
- Improve processes

**Strengthen connectivity**
- Extend standard cloud services
- Think about AUTOSAR App store
- Enable connectivity to smart phones and zone ECUs

**Develop flexible updates during life time**
- Improve modularity e.g. define cluster interfaces
- Provide means for system description
Agenda

- Introduction of the AUTOSAR Partnership
- Challenges in the automotive industry
- Architecture and recent features
- Smart solutions based on AUTOSAR
- Processes and quality
- Outlook
AUTOSAR in a Vehicle Network

Common Bus Interface Specification

- Application Layer
  - RTE
  - AUTOSAR Classic Platform
- Application Layer
  - RTE
  - AUTOSAR Classic Platform
- Application Layer
  - AUTOSAR Adaptive Platform
- Applications
  - Non-AUTOSAR Basic Software
- Applications
  - Non-AUTOSAR Basic Software

e.g. SOME / IP
The Foundation assures compatibility of the different AUTOSAR standards and therefore contains all common artifacts such as …
The Methodology, derived out of the Meta Model, …

… provides means to describe the AUTOSAR architecture with all its interfaces

… defines exchange formats and description templates (e.g. manifest) to enable
  • a seamless integration of the complete vehicle E/E architecture,
  • the automatized configuration of the µC- and µP-software stacks and
  • the seamless integration of application software

… supports means to ensure safety and security of the system

… provides templates to document the standard
The layered architecture of the classic platform basically supports:

- Hardware abstraction
- Scheduling of runnables and tasks (OS)
- Communication between applications on the same hardware and over the network
- Diagnosis and diagnostic services
- Safety- and Security Services
AUTOSAR Classic Platform
Layered Software Architecture (2/2)
Agenda

› Introduction of the AUTOSAR Partnership
› Challenges in the automotive industry
› Architecture and recent features
› Smart solutions based on AUTOSAR
› Processes and quality
› Outlook
Software Architecture – AUTOSAR Defined Interfaces

Use Case ‘Front Light Management’: Exchange Type of Front Light

- SwitchEvent
  - check_switch() (Supplier B)
  - Switch_event(event) (Integrator)

- LightRequest
  - switch event(event) (Supplier B)
  - request_light(type, mode) (Integrator)

- Front-Light Manager
  - request_light(type, mode)
  - get_keyposition()
  - set_light(type, mode)
  - set_dboard(type, mode) (OEM)

- Headlight
  - set_light(type, mode)
  - set_current(…) (Supplier A)

- Standardized Interface
- Std. AUTOSAR Interface
- Services
- Std. Interface

- Standardized Interface
- Std. AUTOSAR Interface
- Communication
- Std. Interface

- AUTOSAR Interface
- ECU Abstraction
- Std. Interface

- AUTOSAR Interface
- CAN Driver

- ECU-Hardware

- DIO
- PWM

- Microcontroller Abstraction

- Complex Drivers

- Supplier B

- Supplier A
Use Case ‘Front Light Management’: Exchange Type of Front Light
## Distribution ECUs

<table>
<thead>
<tr>
<th>SwitchEvent</th>
<th>LightRequest</th>
<th>Front-Light Manager</th>
<th>Xenonlight</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>switch_event</code></td>
<td><code>request_light</code></td>
<td><code>request_light</code></td>
<td><code>set_light</code></td>
</tr>
<tr>
<td><code>(event)</code></td>
<td><code>(type, mode)</code></td>
<td><code>(type, mode)</code></td>
<td><code>(type, mode)</code></td>
</tr>
<tr>
<td>AUTOSAR Int.</td>
<td>AUTOSAR Interface</td>
<td>AUTOSAR Interface</td>
<td>AUTOSAR Interface</td>
</tr>
</tbody>
</table>

```python
def switch_event(event):
    # SwitchEvent

def request_light(type, mode):
    # LightRequest

def request_light(type, mode):
    # Front-Light Manager

def set_light(type, mode):
    # Xenonlight
```

**Example:**
- **SwitchEvent**
  - `switch_event(event)`
- **LightRequest**
  - `request_light(type, mode)`
- **Front-Light Manager**
  - `request_light(type, mode)`
  - `set_light(type, mode)`
- **Xenonlight**
  - `set_light(type, mode)`
  - `set_current(...)`
## Distribution on ECUs – ‘Front-Light Management’

<table>
<thead>
<tr>
<th>SwitchEvent</th>
<th>LightRequest</th>
<th>Front-Light Manager</th>
<th>Xenonlight</th>
</tr>
</thead>
<tbody>
<tr>
<td>check_switch()</td>
<td>switch_event(event)</td>
<td>request_light(type, mode)</td>
<td>set_light(type, mode)</td>
</tr>
<tr>
<td>switch_event(event)</td>
<td>request_light(type, mode)</td>
<td>get_keyposition()</td>
<td>set_current(...)</td>
</tr>
</tbody>
</table>

**AUTOSAR Interface**

**Front-Light Manager**
- request_light(type, mode)
- get_keyposition()
- set_light(type, mode)

**Xenonlight**
- set_light(type, mode)

**AUTOSAR Interface**

**SwitchEvent**
- check_switch()
- switch_event(event)

**LightRequest**
- request_light(type, mode)

**AUTOSAR Interface**

**CAN Bus**
AUTOSAR Platform Application
Continuous improvement cycle for ADAS systems

On Board
- Scene Understanding
- Mission and Trajectory Planning
- Plan Execution, Motion Control
- Sensor Fusion
- Perception

Off Board
- Labeling of trained data
- Evolution by training
- Simulation and sign-off

Secure data exchange on top of DDS SOME/IP or REST
AUTOSAR Platform Application

User Applications

- ara::com Communication Mgmt.
  - RESTful
- ara::time Time Synchronization
- ara::per Persistence
- ara::hms Platform Health Mgmt.
- ara::exec Execution Mgmt.
- ara::idm Identity Access Mgmt.
- ara::log Logging & Tracing
- ara::crypto Cryptography
- ara::s2s service Signal to Service Mapping
- ara::nms service Network Management
- ara::um service Update and Configuration Management

(Virtual) Machine / Container / Hardware

>> Run Execution Manager
... 
... 
... 
ara::exec > start { Vehicle Manager }
The operation completed successfully
ara::exec > start { HMI }
The operation completed successfully
ara::exec > start { Distance Radar }
The operation completed successfully
### AUTOSAR Introduction

**Vehicle Manager**
- ara:com Communication Mgmt.

**HMI**
- ara:rest RESTful
- ara:time Time Synchronization
- ara:per Persistency
- ara:hsm Platform Health Mgmt.
- ara:core Core Types
- ara:exec Execution Mgmt.
- ara:lam Identity Access Mgmt.
- ara:sos Service State Management
- ara:diag service Diagnostics
- ara:adi service Automated Driving Interfaces
- ara:s2s service Signal to Service Mapping
- ara:cm service Network Management
- ara:usem service Update and Configuration Management

**Distance Radar**
- ara:log Logging & Tracing
- ara:crypto Cryptography

**New User Application?**

---

**Vehicle Manager**

**HMI**

**Distance Radar**

**New User Application?**

---

**AUTOSAR Runtime for Adaptive Applications (ARA)**

(Virtual) Machine / Container / Hardware
Adaptive Cruise Control (ACC)
User Applications

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ara::com Communication Mgmt.</td>
<td>RESTful</td>
</tr>
<tr>
<td>ara::rest RESTful</td>
<td>Time Synchronization</td>
</tr>
<tr>
<td>ara::per Persistence</td>
<td>Platform Health Mgmt.</td>
</tr>
<tr>
<td>ara::plm Platform Health Mgmt.</td>
<td>Identity Access Mgmt.</td>
</tr>
<tr>
<td>ara::log Logging &amp; Tracing</td>
<td></td>
</tr>
<tr>
<td>ara::crypto Cryptography</td>
<td></td>
</tr>
<tr>
<td>ara::ucm service Update and Configuration Management</td>
<td></td>
</tr>
<tr>
<td>ara::s2s service Signal to Service Mapping</td>
<td></td>
</tr>
<tr>
<td>ara::sm service Network Management</td>
<td></td>
</tr>
<tr>
<td>ara::adi service Automated Driving Interfaces</td>
<td></td>
</tr>
</tbody>
</table>

**AUTOSAR Runtime for Application (ARA)**

(Virtual) Machine / Container / Hardware

- **HMI**
- **Vehicle Manager**
- **ACC**
- **Distance Radar**

- **Run Execution Manager**
  - `ara::exec > start { Vehicle Manager }`
  - Operation completed successfully
  - `ara::exec > start { HMI }`
  - Operation completed successfully
  - `ara::exec > start { Distance Radar }`
  - Operation completed successfully
  - `ara::exec > start { ACC }`
  - Operation completed successfully
AUTOSAR Introduction
### AUTOSAR Introduction

#### Vehicle Manager
- Communication Mgmt.
- Persistence
- Core Types

#### HMI
- RESTful
- Time Synchronization
- Execution Mgmt.
- Identity Access Mgmt.

#### Distance Radar
- Platform Health Mgmt.
- Logging & Tracing
- Cryptography

#### ACC
- State Service Management
- Diagnostics
- Automated Driving Interfaces
- Signal to Service Mapping
- Network Management
- Update and Configuration Management

**POSIX PSE51 / C++ STL**
- Operating System

**AUTOSAR Runtime for Adaptive Applications (ARA)**

**(Virtual) Machine / Container / Hardware**

**ACC**
Introduction of the AUTOSAR Partnership
Challenges in the automotive industry
Architecture and recent features
Smart solutions based on AUTOSAR
Processes and quality
Outlook
AUTOSAR Adaptive Platform development approach

**Specification**

**Identify needs & use-cases:**
1) Concepts
2) Features
3) Requirements

**Gain speed:**
1) Spec validation
2) Reduce room for spec interpretation
3) Training / dissemination of AP

**Implementation**

**Attracting environment for coders:**
- Appealing technology (C++, Yocto, Git, …)
- Modern use case (ADAS EBA)
- Handy documentation (Wiki)
- Peer programming sessions

**Demonstration**

**Gain trust:**
1) Advertises the progress
2) Highlights some specific features

**Quality:**
- TF-ARC approval
- Cross team review
- Lifecycle: preliminary → draft → valid

**Show AUTOSAR interoperability**
- of classic and adaptive platforms
- but also with others

Best tradeoff between commercial cooperation & compatibility between different vendors
Agenda

- Introduction of the AUTOSAR Partnership
- Challenges in the automotive industry
- Architecture and recent features
- Smart solutions based on AUTOSAR
- Processes and quality
- Outlook