Enabling Connectivity with Service Oriented Architectures

Vector GB Conference 2017
Agenda

Different Perspectives on Connectivity

From Signal Oriented to Service Oriented Architectures
Future E/E Architectures
Service Oriented Architectures with AUTOSAR Classic
Service Oriented Architectures with AUTOSAR Adaptive
Summary and Outlook
Different Perspectives on Connectivity

New Vehicle Functions and Services with Connectivity

- The IT Backend Perspective
- The Vehicle Perspective

- The Business Perspective
- The Technical Perspective

Different Perspectives on Connectivity

- **From Signal Oriented to Service Oriented Architectures**
  - Future E/E Architectures
  - Service Oriented Architectures with AUTOSAR Classic
  - Service Oriented Architectures with AUTOSAR Adaptive
  - Summary and Outlook
From Signal Oriented to Service Oriented Architectures

Characteristics of Signal Oriented Architectures

- Distributed systems with signal-based communication
  - Between SW components and
  - Between ECUs
- Many decisions are done at design time:
  - Binding of SW components
  - Communication design: Signal, PDU, Frame and Schedule Design
- The complete system is known and considered at design time ("Closed static distributed system")
- Distributed SW architecture is based on SW Components with sender/receiver communication
- AUTOSAR Classic is a mature, widely-accepted and used implementation platform
- Implementation in C

Design Workflow for Signal Oriented Architectures in AUTOSAR Classic

Software Architecture

Software Design

Hardware Design

Software/Hardware Mapping

Data Mapping

Signal Routing

Bus Access
- e.g. LIN Schedule

CAN Design*

FlexRay Design

LIN Design

Hardware Architecture

Frame Layout

Import

Export
From Signal Oriented to Service Oriented Architectures

Signal Oriented Architectures - Summary

**Strengths**
- The complete system is known
- The system behaviour is predictable at design time (real-time behaviour and memory resources)
- Optimisations regarding resources are possible at design time (message-based routing, bus scheduling,...)

**Weaknesses**
- Low flexibility at run-time. The system cannot be easily extended after design-time
- High design and test efforts
  - For communication (Signals, PDUs, Frames and Schedules)
  - For start-up and shut-down behaviour of the distributed system (network management, ...)
- No support for more complex SW architectures (SW components with only Sender/Receiver Interfaces are used)

**Customer Benefits**
- Predictability of distributed embedded real-time systems
- Cost-optimised implementation with low resources for microcontrollers and communication technologies
Characteristics of Service Oriented Architectures

- Distributed systems with service-based communication
  - Between SW components and
  - Between ECUs
- Important decisions are done at run-time, not at design-time!
  - Binding (by service discovery)
  - Communication is dynamic (e.g. dynamic PDU content transmitted using serialisation and deserialisation)
- "Open system": Services can be added at run-time (e.g. outside the vehicle)
- The SW Architecture is designed with SW components with service interfaces
- AUTOSAR Adaptive as implementation platform currently in specification
- Implementation in C++

... but AUTOSAR Classic can also be used as an implementation platform for Service Oriented Architectures
What is a Service?

Service Interface
- Method
- Fire&Forget Method
- Property
- Event

Service Contract
- Service Provider Port
- Service Consumer Port

Technology Mapping

AUTOSAR Adaptive ECU/System
- Service Provider
- Service Consumer
- Switch
- SWC1
- SWC2
- SWC3

Service Discovery
- Serialisation/Deserialisation

Ethernet

AUTOSAR Classic ECU/System
- Service Provider
- Service Consumer
- Switch
- SWC1
- SWC2
- SWC3

Technology Mapping
- Receiver, Sender, Client, Server Ports

Service Discovery
- Serialisation/Deserialisation

Ethernet
What is a Service?

Service Participant 2

Service Participant 3

Service Interface
- Method
- Fire&Forget Method
- Property
- Event

Service Contract

Service Participant 2
- Service Provider Port

Service Participant 3
- Service Consumer Port

SOA Diagram in PREEvision 8.0
From Signal Oriented to Service Oriented Architectures

Service Interface and Technology Mapping to AUTOSAR Classic

1: Fire and Forget Method = Method without return
2: Property = Field = Attribute
Design Workflow for Service Oriented Architectures in AUTOSAR Classic

From Signal Oriented to Service Oriented Architectures

**Service Design**
- SWC 1
- SWC 2
- SWC 1
- SWC 2

**Software/ Hardware Mapping**
- SWC 1
- SWC 2

**Signal Routing**
- Data Mapping, Data Transformation /Serialization
- Switch Configuration, VLAN
- Communication Design, Socket Configuration

**Hardware Design**
- ECU 1
- ECU 2
- ECU 3

**Software Design**
- Service Design
- My Class
  - Property 1
  - Method 1 (9)
  - Method 2 (9)

**Communication**
- Import
- Export

**Table**

<table>
<thead>
<tr>
<th>Port</th>
<th>VLAN</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>T</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>T</td>
<td>U</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4094</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

T = send tagged, U = send untagged
From Signal Oriented to Service Oriented Architectures

Service Oriented Architectures: Summary

**Strengths**
- Higher flexibility at run-time
- Services inside and outside the vehicle can be linked
- Extensions and updates of services are possible
- Simplified communication design
- Simplified start-up and shut-down design
- Implementation of complex and open SW architectures (Complex Interfaces, Layers, Remote Procedure Calls, ...) possible
- Maintenance and extensions of SW architectures are easier

**Weaknesses**
- Predictability of system behaviour (different results of service discovery possible)
- Higher resources for implementation needed (microcontrollers and communication technologies)
- Many different, dynamic system configurations have to be tested

**Customer Benefits**
- Flexible, open and dynamic distributed embedded systems
Service Oriented Architectures are Enablers …

- **for Connectivity**
  - Completely new functions by flexible integration of services in the IT backend

- **for Autonomous Driving**
  - Communication from the vehicle to the IT backend and vice versa
  - Creation of precise maps
  - Predictive assistance based on precise maps
  - Routing for vehicle fleets in the IT backend
  - …

- **for SW Update, SW Upgrade and Service for Vehicles in the Field**
  - SW update of single services
  - SW update and upgrade for vehicles in the field Over The Air
  - Remote diagnostics
  - …

- **for Variant and Building Set Strategies**
  - SW options can be implemented as services (base service ... premium service)
Agenda

Different Perspectives on Connectivity
From Signal Oriented to Service Oriented Architectures

Future E/E Architectures
- Service Oriented Architectures with AUTOSAR Classic
- Service Oriented Architectures with AUTOSAR Adaptive
Summary and Outlook
Future E/E Architectures

Signal Oriented versus Service Oriented Architectures

- **Strengths of Service Oriented Architectures**
  - Higher flexibility at run-time
  - Services inside and outside the vehicle can be linked
  - Extensions and update of services are possible
  - Simplified communication design
  - Simplified start up and shut down design
  - Implementation of complex and open SW architectures (Complex Interfaces, Layers, Remote Procedure Calls, …) is possible
  - Maintenance and extensions of SW architectures are possible

- **Strengths of Signal Oriented Architectures**
  - The complete system is known
  - The system behavior is predictable at design time (real time behavior and memory resources)
  - Optimisations regarding resources are possible at design time (message-based routing, bus scheduling,…)
  - Many years of experience
  - Many mature components exist

- **Customer Benefits**
  - Flexible, open and dynamic distributed embedded systems

- **Customer Benefits**
  - Predictability of distributed embedded real time systems
  - Cost optimised implementation with low resources for microcontrollers and communication technologies
Future E/E Architectures will combine the strengths of both:

- Signal Orientation and Service Orientation
- "Hybrid architectures" are expected
Future E/E Architectures

Signal Oriented or Service Oriented Architectures?

"Hybrid ECUs" are expected
Agenda

Different Perspectives on Connectivity
From Signal Oriented to Service Oriented Architectures
Future E/E Architectures

Service Oriented Architectures with AUTOSAR Classic
Service Oriented Architectures with AUTOSAR Adaptive
Summary and Outlook
Service Oriented Architectures with AUTOSAR Classic

**SOA Use Cases**

- Introduction of Service Oriented Architectures
- Migration from Signal to Service Oriented Architectures
- Design of Hybrid Architectures in AUTOSAR Classic

**PREEvision Layers**

- Requirements
- Logical Function Architecture
- Software/Service Architecture
- Hardware Architecture
- Wiring Harness

**Software/Service Architecture**

- Service Oriented Architecture
- SW Library
- SWC Instances
- AUTOSAR Classic

**Import/Export**

- Import
- Export
Service Oriented Architectures with AUTOSAR Classic

SOA Design Workflow
Migration from Signal to Service Orientation

Signals:
- WheelSpeed_FR (Signal)
- WheelSpeed_FL (Signal)
- WheelSpeed_RR (Signal)
- WheelSpeed_RL (Signal)

... to Service Orientation:

Services:
- WheelSpeeds (Service Interface)
  - WheelSpeed_FR (Event)
  - WheelSpeed_FL (Event)
  - WheelSpeed_RR (Event)
  - WheelSpeed_RL (Event)
Migration from Signal to Service Orientation

Service Oriented Architectures with AUTOSAR Classic

© Dr. Matthias Traub, BMW: Mitten im Umbruch zur Digitalisierung.
Presentation on the 8. Vector Congress, Nov. 29.-30, 2016, Stuttgart
Service Oriented Architectures with AUTOSAR Classic

The complete Vector AUTOSAR Tool Chain

System Design
- Architecture and Communication Design: PREEvision
- Diagnostics Design: CANdelaStudio

Application Software Development
- SWC Design: PREEvision, DaVinci Developer
- SWC Execution and Test: vVIRTUALtarget pro

ECU SW Integration
- BSW/RTE Configuration: DaVinci Configurator Pro
- Virtual Integration: vVIRTUALtarget basic

AUTOSAR ECU
- Basic Software (BSW)
- RTE

AUTOSAR ECU/System
- SWC1
- SWC2
- SWC3

AUTOSAR ECU/System

System Verification
- ECU Calibration: CANape
- Verification of Network Communication and Diagnostic Behavior: CANoe & CANoe.DiVa

Application Software Verification
- SWC Verification within Real ECU: CANoe & VT System
- SWC Verification in Virtual Environment: vVIRTUALtarget pro

ECU SW Verification
- ECU Monitoring and Debugging: CANoe.AMD
Service Oriented Architectures with AUTOSAR Classic

AUTOSAR at the System Level

AUTOSAR Workflow Support
Agenda

Different Perspectives on Connectivity
From Signal Oriented to Service Oriented Architectures
Future E/E Architectures
Service Oriented Architectures with AUTOSAR Classic

- **Service Oriented Architectures with AUTOSAR Adaptive**

Summary and Outlook
Service Oriented Architectures with AUTOSAR Adaptive

AUTOSAR Adaptive Support

PREEvision Layers

- Requirements
- Logical Function Architecture
- Software/Service Architecture
- Hardware Architecture
- Wiring Harness

Software/Service Architecture

- Service Oriented Architecture
- SW Library
- SWC Instances
  - AUTOSAR Classic
  - AUTOSAR Adaptive

Import Export

AUTOSAR Classic

AUTOSAR Adaptive

Manifest
Service Oriented Architectures with AUTOSAR Adaptive

UML Diagrams

**AUTOSAR Classic Platform**

- **WheelSpeed** (Receiver Port)
- **WheelPressure** (Client Port)
- **VehicleSpeed** (Sender Port)
- **WheelState** (Server Port)

**AUTOSAR Adaptive Platform**

- **Service Consumer Port**
- **Service Provider Port**

**VehicleSpeedInterface** (Service Interface)
- **WheelSpeed** (Event)
- **WheelPressure** (Method)

**WheelInterface** (Service Interface)
- **WheelSpeed** (Event)
- **WheelPressure** (Method)

**Needed UML Diagrams:**
- Use Case Diagram
- SOA Diagram
- Class Diagram
- Sequence Diagram
- State Diagram

**Software/Service Architecture**

1. Service Oriented Architecture
2. SW Library
3a. SWC Instances AUTOSAR Classic
3b. SWC Instances AUTOSAR Adaptive

**Service Interface (Class)**
- Property (Attribute)
- Method (Operation())
- Method Fire&Forget (Operation())
- Event (Operation())
Different Perspectives on Connectivity
From Signal Oriented to Service Oriented Architectures
Future E/E Architectures
Service Oriented Architectures with AUTOSAR Classic
Service Oriented Architectures with AUTOSAR Adaptive

Summary and Outlook
Enabling Connectivity with Service Oriented Architectures

Software/Service Architecture

Service Oriented Architecture

SW Library

SWC Instances

AUTOSAR Classic

AUTOSAR Adaptive

SW Library

AUTOSAR Classic

AUTOSAR Adaptive

Common Bus Interface Specification

Application SW-C

AUTOSAR Classic Platform

RTE

Application SW-C

AUTOSAR Classic Platform

RTE

Adaptive Application SW-C

Adaptive Application SW-C

AUTOSAR Adaptive Platform

SWC Non AR

non AUTOSAR Basic Platform

non AUTOSAR Offboard System

Enabling Connectivity with Service Oriented Architectures

- Service Oriented Architectures (SOAs) provide flexible, open and dynamic distributed systems
- They are enablers for
  - Connectivity and Autonomous Driving
  - SW Update, SW Upgrade and Vehicle Service in the field
  - New Building Set Strategies and Handling of Variants
- Future E/E Architectures will combine the strengths of Signal Oriented and Service Oriented Architectures
- AUTOSAR Adaptive and AUTOSAR Classic will be deployed in the same vehicle, even in the same ECU

- The introduction of SOAs, the migration to SOAs and their implementation in AUTOSAR Classic are the first steps

- The next steps are:
  - Implementation in AUTOSAR Adaptive,
  - Hybrid applications of AUTOSAR Adaptive and AUTOSAR Classic, and
  - SOA Design with UML Class, Sequence, and State Diagrams
Your questions are welcome!

Author: Jörg Schäuffele & Iain Cunningham
Vector Germany