Model Service-Oriented Architectures (SOA) in Simulink

Luigi Milia – Automotive Industry Manager
Shwetha Bhadravathi Patil – AUTOSAR Product Manager
Agenda

- SOA: a new paradigm for automotive software
- Simulink for Service-Oriented Applications
- Key take-aways
Agenda

- SOA: a new paradigm for automotive software
- Simulink for Service-Oriented Applications
- Key take-aways
Your next car will have a lot more software

- **How to** add more software
  - Add more functions to ECUs?
  - Add more ECUs to E/E network?

- Incremental approach **doesn’t scale**...

> To ensure **safety** in increasing degrees of **autonomy**, software **quality** and **complexity** are a key challenge for the automotive industry, requiring a **rethink** today’s vehicle **software and E/E architectures**.*

* SDV Trends, Challenges, and Implications for OEMs - McKinsey presentation at Gasgoo – July 2020
Industry is investing to transform software development

- **Consolidation** of people and electronics
- **Development of new software platforms**

---

Bosch Consolidates All Automotive Software And Electronics Into New Division

The complexity of modern electronics/electrical (E/E) architectures is making it more challenging to actually assemble components like wiring harnesses into vehicles so suppliers like Bosch and competitors Continental and Aptiv are following the path set out by Tesla with more powerful computers that consolidate the capabilities of many smaller discrete electronic control units. This in turn can help lead to simplified wiring.

---

OEMs developing their own **vehicle operating systems**: VW.OS from **Car.Software Org**

---


Evolution of E/E architectures

**yesterday**
- Centralized gateway
- ~100 ECUs
- One function per ECU
- Heavy wiring

**today**
- Domain controllers (powertrain, chassis, body, ...)
- Multi-core ECUs
- Consolidation of functions

**tomorrow**
- Vehicle/zone controllers (front, rear, ...)
- High-performance CPUs (Multi-core, GPU, FPGA)
- High-speed ethernet
Towards Service-Oriented Architectures

yesterday
- High SW-HW coupling
- No standard APIs
- No/minimal SW reuse

Software

Hardware

today
- Static SW component allocation (design-time)
- Signal-based communication
- HW abstraction
- Monolithic update (full image flashing)

Application Software

RTE

Basic Software

Hardware

Application Services

Basic Services

Middleware

High Performance Hardware/Virtual Machine

Legacy

Components-based

Service-oriented

tomorrow
- Dynamic Service discovery (run-time)
- Service-oriented communication
- Higher HW abstraction
- Selective updates (OTA)
Simulink: design software once, deploy to many targets

Legacy ECU

- μC
- FPGA
- GPU

AUTOSAR Classic

- Application Software
- RTE
- Basic Software

AUTOSAR Adaptive / ROS / DDS

- Application Services
- Basic Services
- Middleware
Agenda

- SOA: a new paradigm for automotive software
- Simulink for Service-Oriented Applications
- Key take-aways
You can model service-oriented communication using messages (Send/Receive).
Simulink support for AUTOSAR Adaptive

- In AUTOSAR Adaptive, services implement **communication** through:
  - Events
  - Methods
  - Fields

- In Simulink, **Events** can be modeled as **Messages** and then configured for code generation using **AUTOSAR Blockset**.

AUTOSAR Adaptive C++ compliant code is generated by Embedded Coder.
Adaptive SW architecture concepts

```
"Radar" : {
   // events
   "event" : {
      "leftLaneDistance"
      "leftTurnIndicator"
      "leftCarInBlindSpot"
      "rightLandDistance"
      "rightTurnIndicator"
      "rightCarInBlindSpot"
   },
   // methods
   "method" : {
      "Calibrate"
      "Adjust"
   },
   // fields
   "field" : {
      "updateRate"
   }
}
```

```
"Hazard" : {
   // events
   "event" : {
      "leftHazardIndicator"
      "rightHazardIndicator"
   },
   // methods
   "method" : {
   },
   // fields
   "field" : {
   }
}
```
Modelling an AUTOSAR Adaptive application in Simulink

```json
"Radar": {
  // events
  "event": {
    "leftLaneDistance",
    "leftTurnIndicator",
    "leftCarInBlindSpot",
    "rightLaneDistance",
    "rightTurnIndicator",
    "rightCarInBlindSpot"
  },
  // methods
  "method": {
    "Calibrate",
    "Adjust"
  },
  // fields
  "field": {
    "updateRate"
  }
}
```
Modelling an AUTOSAR Adaptive application in Simulink

```json
"Hazard" : {
  // events
  "event" : {
    "leftHazardIndicator",
    "rightHazardIndicator"
  },
  // methods
  "method" : {},
  // fields
  "field" : {}
}
```
Dynamic Service Discovery

Find adaptive services by using dynamic discovery

- Configure AUTOSAR adaptive applications to **discover and subscribe to adaptive services** as they become available

- You can also configure service port programmatically as OneTime or DynamicDiscovery

```java
apiObj = autosar.api.getAUTOSARProperties("autosar_LaneGuidance");
apiObj.set("/LaneGuidance_pkg/LaneGuidance_swc/LaneGuidance/RequiredPort/",
  "ServiceDiscoveryMode", "DynamicDiscovery")
```
AUTOSAR Adaptive workflows

Top-Down

Export ARXML → AUTOSAR SW-C Description → Import ARXML

Bottom-Up

Import ARXML → Application SW C++ Code → Embedded Coder → Simulink, AUTOSAR Blockset
AUTOSAR Adaptive in action

- Create model from ARXML
- Verify AUTOSAR properties
- Configure Service Discovery
- Generate code
Agenda

- SOA: a new paradigm for automotive software
- Simulink for Service-Oriented Applications
- Key take-aways
Key take-aways

- Automotive E/E and SW architectures are changing dramatically

- Service-oriented architectures are foundation for dynamic SW configuration, updates and event-driven communication

- Leverage the power of Model-Based Design to model, simulate and deploy SOA applications compatible with AUTOSAR Adaptive, ROS and DDS.
Additional resources

- Learn more:
  - https://www.mathworks.com/discovery/soa.html

Examples and How To

- Message-Based Communication Between Software Components - Example
- Run-Time Software Modeling (5 Videos) - Video Series
- Model AUTOSAR Adaptive Software Components - Example
- Configure AUTOSAR Adaptive Service Communication - Example
Presenter contact info and poll questions

Please contact us with questions

Luigi Milia
Automotive Industry Manager
lmilia@mathworks.com

Shwetha Bhadravathi Patil
AUTOSAR Product Manager
shwethap@mathworks.com

▪ Are you already working on SOA based applications?

▪ Do you plan to work on SOA based applications in the near future?

▪ Do you plan to use MBD for SOA?

▪ Are you interested in a follow-up conversation with MathWorks?