10th AUTOSAR Open Conference

Rinat Asmus
AUTOSAR Project Leader
BMW

AUTOSAR Adaptive Platform
New Challenges – One Standard.

Do AUTOSAR Core partners **commit** to standard use?

Is the implementation of the standard **applicable**?

Does AUTOSAR provide the **functionality** my software application requires?

AUTOSAR Runtime for Adaptive Applications (ARA)

- Execution Management
- Identity Access Management
- Time Synchronisation
- Update and Configuration Management
- Diagnostics
- Operating System
- Communication Management
- Persistency
- Signal 2 Service Mapping
- Network Management
- Not part of AUTOSAR standardization.
- Platform Health Management
- Logging and Tracing
- Cryptography
- REST

ARA includes POSIX profile PSE51.
Additional APIs are extended by the functional clusters.

Adaptive Platform Services
Adaptive Platform Foundation

Bootloader

Machine
Service vs. API implementation.

AUTOSAR Runtime for Adaptive Applications (ARA)

Adaptive Platform Foundation
.. provides detached additional features
- implemented as C++ libraries
- Communicates over inter process communication directly

Adaptive Platform Services
.. are coupled to ara middleware implementation
- C++ code is generated through information passed by to ara::com generator
- Onboard communication is managed by ara::com

Machine

Execution Management
... schedules processes and manages the machine states

Diagnostics
... to manage software errors and reaction
ARA COM middleware.

- SoA to intra and inter machine communication
- Generates Proxy/Skeleton as defined in Methodology
- Supports static (preconfigured) and dynamic (during runtime) service discovery
- Specifies additional data types such as collection types or function wrappers
- Abstracts from network communication protocol layer
- ARA API is designed for event-driven and polling approaches
- supports a seamless integration of end2end protection
Features of Adaptive Platform Release 17-03.

- **METHODOLOGY**
  - Diagnostic Extract
  - Manifests
  - Methodology Extensions for Adaptive

- **COMMUNICATION**
  - Service-Discovery and Service-based Communication
  - Events and Methods
  - Language-Binding: C++
  - Inter-Process-Communication
  - Bus-Binding: SOME/IP

- **POSIX OS**
  - OS Application Interface (PSE51 & C++STL)

- **EXECUTION MANAGEMENT**
  - Integration of Applications onto Platform
  - Start and Stop of Applications

- **DIAGNOSTICS**
  - DTC Management
  - ISO 14229 / ISO 13400

- **PERSISTENCY**
  - Data Storage

- **SAFETY**
  - C++14 Coding Guideline
  - E2E Communication Integrity

- **LOGGING / TRACING**
  - Logging and Tracing

**METHODOLOGY**
- Methodology Extensions for Adaptive
- Service to signal modeling
- Predefined Data Types

**EXECUTION MANAGEMENT**
- Resource Management
- Recovery action framework
- Machine State handling

**COMMUNICATION**
- Time Synchronization
- Support of RESTful
- Fields

**SAFETY**
- C++14 Coding Guideline
- E2E for periodic communication

**DIAGNOSTICS**
- ISO 13400 / ISO 14229 completion
- Handling of SW Clusters

**SECURITY**
- Crypto API
- Authentication & Certificates
- Key Management
- Secure Communication
Features of Adaptive Platform Release 18-03.

**METHODOLOGY**
- Methodology Extensions for Adaptive
- Service to signal modeling

**EXECUTION MANAGEMENT**
- Resource Management
- Parallel Processing by HWA
- Recovery action framework

**SW-CONFIGURATION MGNT**
- Package Management
- Installation routine

**COMMUNICATION**
- Support of RESTful
- Network Management Ethernet

**DIAGNOSTICS**
- Maintenance and Improvements

**PERSISTENCY**
- Data Storage
- Safe Data Storage
- Persistent Data Encryption

**SAFETY**
- Platform Health Management
- Safety concept for AP
- E2E for non periodic communication

**SECURITY**
- Crypto API
- Authentication & Certificates
- Key Management
- Secure Communication
- Support of trusted platform

**LOGGING / TRACING**
- Maintenance and Improvements

**EXECUTION MANAGEMENT**
- Resource Management
- Parallel Processing by HWA
- Recovery action framework

**R18-03**
Specification and Validation by Implementation.

The Adaptive Platform is validated through an AUTOSAR-internal implementation: The Adaptive Platform Demonstrator

This Demonstrator is available to all the partners and can be a reference to understand the underlying concepts of the Adaptive Platform.

All further development based on the Demonstrator will become the responsibility of the respective partner.
Join AUTOSAR.

- **WP-A** Software Architecture
  - WP-A-LIB Libraries
  - WP-A-MCBD Multicore BSW Dist

- **WP-A1** VFB and RTE

- **WP-A2** COMM Stack

- **WP-A4** Diagnostics

- **WP-A5** MCAL

- **WP-M** Methodology and Templates
  - WP-M-METH Methodology
  - WP-M-GST Generic Structure Template
  - WP-M-SWCT Software Component Template
  - WP-M-SYST System Template ECU Configuration
  - WP-M-TIMEX Timing Extensions

- **WP-I** Application Interfaces
  - WP-I-BODY Body and Comfort
  - WP-I-ENGINE Powertrain Engine
  - WP-I-TRSM Powertrain Transmission
  - WP-I-CHASSIS Chassis Control
  - WP-I-OCSAFE Occupant and Pedestrian Safety

- **WP-M1** Timing Analysis

- **FT-CM** Communication Management
  - FT-EMO Execution Management
  - FT-DIA Adaptive Diagnostics
  - FT-MM Methodology & Manifests
  - FT-PER Persistency
  - FT-UCM Update & Conf Management

- **FT-SEC** Security
  - FT-ADI Automated Driving Interfaces
  - FT-ST System Tests

- **FT-ADI** Demonstrator Integration
  - FT-UCM Update & Conf Management

- **Legend:**
  - Lead Work Package
  - Subgroups
  - Feature Team
  - Work Package

New in 2018
Thank you for your attention!

Rinat Asmus
Hufelandstr. 1
80937 Munich, Germany
Tel: +49-89-382-65843
Mail: rinat.asmus@bmw.de