Document Title	Classic Platform Release Overview
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	0

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R19-11

Document Change History			
Date	**Release**	**Changed by**	**Description**
2019-11-28	R19-11	AUTOSAR Release Management	Release Life Cycle Status: R19-11 is in Evolution, R19-11 supersedes R4.4

Disclaimer

Table of Contents

References

[1] SAE J1939-81 Network Management

1 Introduction

1.1 Scope of this document

This document provides an overview of the AUTOSAR standard Classic Platform Release R19-11.

1.2 AUTOSAR standards

1.2.1 Introduction

AUTOSAR addresses a wide range of use cases in automotive software development with its standards. These use cases have different requirements and lead to different technical solutions.

Packaging its deliverables into different "standards"

- eases the access to AUTOSAR solutions for users and
- allows AUTOSAR to scale with market needs.

1.2.2 Definition

An AUTOSAR standard is a consistent set of AUTOSAR deliverables, which are released at the same time. AUTOSAR deliverables can, but are not limited to be of the following kinds:

- textual explanations
- textual specifications
- test specifications
- source code
- other formal or semi-formal textual formats (e.g. ARXML, UML models, XML Schemata)

At the time of release, AUTOSAR ensures that dependencies are fulfilled.

1.2.3 Overview on AUTOSAR's standards

AUTOSAR delivers the following standards:

Standard	Abbreviation
Adaptive Platform	AP
Classic Platform	CP
Foundation	FO

1.2.3.1 Adaptive Platform

The Adaptive Platform is AUTOSAR's solution for high-performance computing ECUs to build safety-related systems for use cases such as highly automated and autonomous driving.

1.2.3.2 Classic Platform

The Classic Platform is AUTOSAR's solution for embedded systems with hard real-time and safety constraints.

1.2.3.3 Foundation

The purpose of the Foundation standard is to enforce interoperability between the AUTOSAR platforms.

Foundation contains common requirements and technical specifications (e.g. protocols) shared between the AUTOSAR platforms.

1.2.4 Dependencies between Standards

Each release of Classic and Adaptive Platform relies on a dedicated version of Foundation. The specific dependency is documented in chapter 1.3.6.

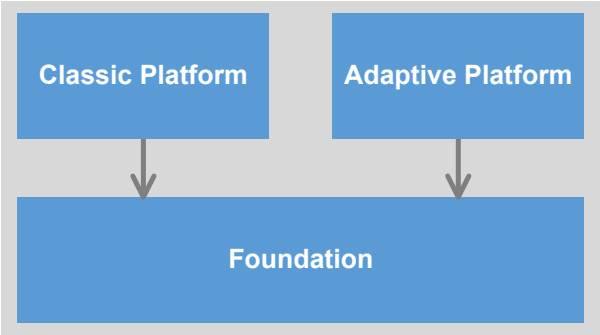

Figure 1.1: Dependencies of AUTOSAR Standards

1.2.5 Dependencies to other standards

This release of the Classic Platform depends on the standard Foundation in release R19-11, which

- defines protocols implemented by Classic Platform
- contains the project objectives and the common requirements from which the features of the Classic Platform are derived
- contains common specification parts which apply to both, the Adaptive Platform and the Classic Platform.

These dependencies are refined in the trace information of the requirements in the respective specifications.

1.3 Release Numbering and Life Cycle

1.3.1 Platform release number

AUTOSAR applies a four-digit numbering scheme Ryy-mm to identify releases. The identifiers “yy” and “mm” depict the year and month of the release date, e.g. R19-11 for the November 2019 release.

1.3.2 Internal release number

AUTOSAR additionally maintains an internal release number for different purposes (e.g. usage in BSW modules in Classic Platform).
The internal release number is used for all platforms and follows up on the Classic Platform release number. In Adaptive Platform this is newly introduced. In Foundation this leads to a discontinuation of the former numbering pattern (e.g. R1.5.0).
A mapping list between Platform Releases and corresponding internal release numbers can be found in chapter 1.3.5. The internal release number uses a three-digit numbering scheme R<major>.<minor>.<revision> to identify releases. Its primary purpose is to identify a release as

- a major release: Valid and draft specification parts may be changed backward incompatibly.
- a minor release: Valid specification parts may only be changed backward compatibly. Draft specification parts may be changed backward incompatibly.
- a revision: Does not contain extensions but only backward compatible bugfixes.

1.3.3 Release life cycle of a major release

Each major release goes through four consecutive steps within its life cycle (examples based on the internal release numbering scheme):

1. Development: Between start of life cycle and the initial release (e.g. R4.0.1)
2. Evolution: Following the initial release with zero, one or several minor releases and/or revisions (e.g. R4.0.2, R4.1.1)
3. Maintenance: No new content is added to a major release but only maintenance of the existing content with zero, one or several revisions (e.g. R3.2.2) is provided
4. Issue Notice: No more revisions but zero, one or several issue notices, i.e. updates of the list of known issues until end of life cycle.

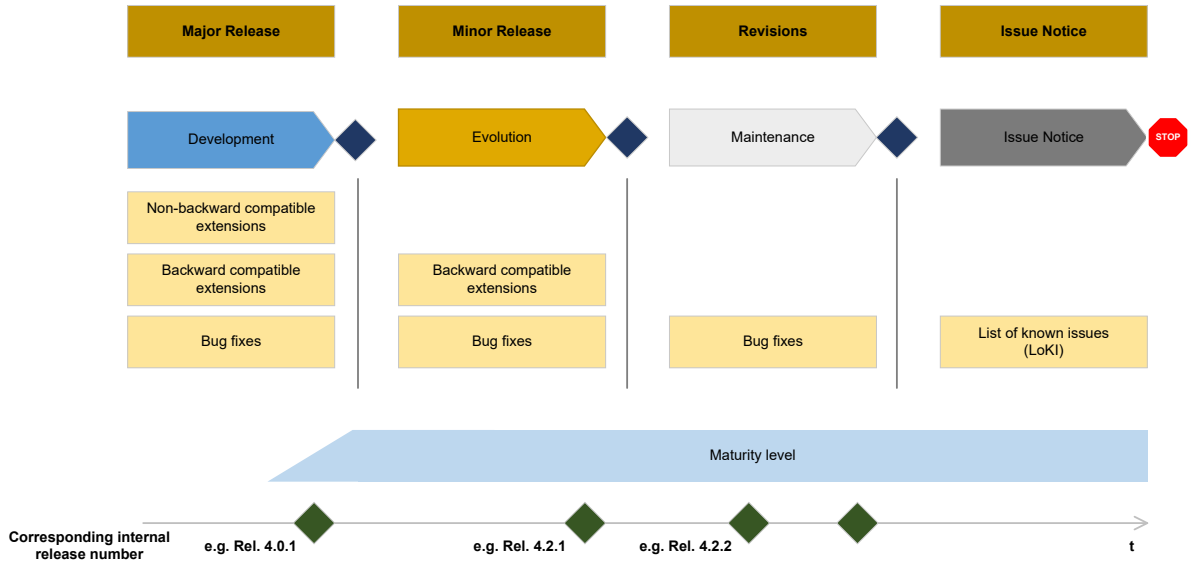

Figure 1.2: Life cycle model of AUTOSAR standards

1.3.4 Life cycle states of specification items and requirements

The life cycle state of a specification item is found after the specification item ID surrounded by curly brackets. The states are:

- {Valid}: This indicates that the related entity is a valid part of the document. This is the default and also applies if no dedicated life cycle status is annotated for the related entity.
- {Draft}: This indicates that the related entity is newly introduced but still experimental. This information is published but is subject to change without backward compatibility guarantee.

- {Obsolete}: This indicates that the related entity is subject to be removed in one of the following releases without further notice.

The life cycle state of a requirement is found in the attribute "type". The states are the same as the specification item states.

1.3.5 Overview of AUTOSAR schema versions and corresponding internal AUTOSAR releases

Schema Version	Platform release	Internal release number
AUTOSAR_00048	R19-11	R4.5.0

According to the release life cycle of AUTOSAR the release R19-11 is a minor release.

1.3.6 Overview of AUTOSAR schema versions and corresponding valid AUTOSAR releases

The AUTOSAR schema does not have an impact on the Foundation. The Foundation releases are mentioned for the sake of completeness.

Schema Version	Classic Platform release	Adaptive Platform release	Foundation release
AUTOSAR_00042	R4.3.0	R17-03	R1.1.0
AUTOSAR_00043	R4.3.0	R17-10	R1.2.0
AUTOSAR_00044	R4.3.1	R17-10	R1.3.0
AUTOSAR_00045	R4.3.1	R18-03	R1.4.0
AUTOSAR_00046	R4.4.0	R18-10	R1.5.0
AUTOSAR_00047	R4.4.0	R19-03	R1.5.1

Schema Version	AUTOSAR release
AUTOSAR_00048	R19-11

1.4 Content of chapters

This document is structured as follows:

- Chapter 1 provides an introduction to AUTOSAR's release strategy and its standardization approach.
- Chapter 2 provides a summary of changes since the previous release of the Classic Platform.

- Chapter 3 contains the overview of specifications comprising the AUTOSAR release R19-11. This chapter is structured according to the clusters of AUTOSAR release R19-11.
- Chapter 4 contains remarks about known technical deficiencies.
- Chapter 5 contains the detailed revision history of all released specifications.

2 Summary of changes

This chapter contains a summary of changes which were implemented since the previous release R4.4.0.

2.1 Release R19-11

Several concepts affecting solely the Classic Platform have been introduced with release R19-11 thereby adding functionalities to the platform (BSW Multicore Distribution, Non-Volatile Data Handling Enhancements, Firmware over the Air).

Additionally some concepts target the Classic and Adaptive Platform, strengthening the interaction between the two platforms.

Those concepts are related to security (IPSec Protocol), communication (Signal Service Translation) and diagnostics (DoIP Extension).

The AUTOSAR XML Schema requires the xml namespace definition file xml.xsd as xsd:import. This file is not released with the AUTOSAR specifications but can be downloaded from `https://www.w3.org/2001/03/xml.xsd`.

2.1.1 Concepts

2.1.1.1 Introduced Concepts

The following concepts in 2.1.1.1.1 - 2.1.1.1.7 have been introduced.

2.1.1.1.1 IPsec Protocol

The concept provides the ability to configure authenticated and/or encrypted communication between ECUs based on the the IETF IPsec standards. Since it works on the IP network layer 3, it can be used transparently. Applications need not to be changed or even be aware that their communication is secured.

2.1.1.1.2 Signal Service Translation

The goal of this concept is to make Adaptive Machines interact with Classic ECUs. Adaptive Platform restricts communication to Service-oriented communication, whereas a major part of the vehicle's ECUs still uses Signal-based communication.

- A translation of these two communication approaches has to be performed:

  - Signal-to-service translation
  - Service-to-signal translation
  - To be implemented on Classic or Adaptive
  - Support for end-to-end safety and security

To support end-to-end safety (E2E) for signal service translation both directions - signal-to-service-translation and service-to-signal-translation - are covered. This includes use cases where both sides of communication use the same E2E profile as well as uses cases where different E2E profiles are in place. While performing signal-to-service translation the E2E status of the received payload is checked and forwarded to the targeted receiving communication part together with the translated payload.

2.1.1.1.3 BSW Multicore Distribution

- Distribute communication stack across multiple microcontroller cores, based on network types
- PduR as central inter-core dispatcher
- Enable load balancing
- Minimize inter-core communication
- Define a 'Basic Software Multicore Library' which provides efficient lock-free implementations for atomic operations

2.1.1.1.4 DoIP Extension

The concept extends the DoIP specification with the possibility to allow diagnostic communication between DoIP nodes and internal testers located within the vehicle network.

2.1.1.1.5 ServiceVersioningARAcom

Support of contract service versioning. The service discovery can be configured to support version backwards-compatibility.

2.1.1.1.6 Non-Volatile Data Handling Enhancements

- Improve support for large(r) NvData items by writing back only changed parts
  - More fine grained control of which data has changed
  - Allow for Structs or Arrays to span multiple NvRam blocks

- Avoid unnecessary data copies in case of implicitly accessed NvRam
  - Provide the necessary information to the RTE
- Enable parameters which are stored in code flash to be modified through DCM Diagnostic Services

2.1.1.1.7 Firmware over the Air (FOTA)

FOTA standardizes the configuration and implementation of remote SW updates over the air. The concept provides guidelines and best practices on how to realize the FOTA features to manage update over the air during run time.

2.1.1.2 Impact of Concepts

The introduced concepts have impact on several specifications. The following table provides a detailed overview.

Please note that some of the specifications are marked by special text formatting:

- Specifications in **bold** font are completely new specifications originating from the particular concept.
- Specifications in *italic* font are affected indirectly as they provide artifacts for the actually impacted specifications.

Concept Name	Specification Long Name	Standard
IPsecProtocol	Requirements on IPsec Protocol	Foundation
	Requirements on AUTOSAR Features	Classic Platform
	System Template	
	Requirements on Ethernet Support in AUTOSAR	
	Specification of TCP/IP Stack	
	Specification of Manifest	Adaptive Platform
Signal Service Translation	Explanation of Foundation Diagram Source	Foundation
	E2E Protocol Specification	
	Main Requirements	
	Glossary	
	General Specification on Transformers	Classic Platform
	Basic Software UML Model	
	Specification of SW-C End-to-End Communication Protection Library	
	Specification of Module E2E Transformer	

▽

△

Concept Name	Specification Long Name	Standard
	Specification of RTE Software	
	Software Component Template	
	System Template	
	Specification of SOME/IP Transformer	
	Specification of COM Based Transformer	
	Layered Software Architecture	
	Requirements on System Template	
	General Specification on Transformers	Adaptive Platform
	Requirements on Manifest Specification	
	Specification of Manifest	
	Requirements on Communication Management	
	Specification of Communication Management	
BSW Multicore Distribution	Guide to BSW Distribution	Classic Platform
	Specification of ECU Configuration	
	Specification of ECU Configuration Parameters (XML)	
	Specification of Communication	
	Specification of Communication Manager	
	Specification of I-PDU Multiplexer	
	Specification of Network Management Interface	
	Specification of PDU Router	
	Specification of RTE Software	
	Specification of Secure Onboard Communication	
	Specification of Synchronized Time-Base Manager	
	Specification of Ethernet Driver	
	Specification of Ethernet Interface	
	Specification of Bus Mirroring	
	Specification of Basic Software Multicore Library	
	Layered Software Architecture	
	Basic Software UML Model	
	Specification of Basic Software Multicore Library	
DoIPExtension	Specification of Diagnostic over IP	Classic Platform
	System Template	
	Specification of ECU Configuration Parameters (XML)	

▽

△

Concept Name	Specification Long Name	Standard
	Basic Software UML Model	
	Requirements on Diagnostics	Foundation
ServiceVersioningARAcom	Specification of Service Discovery	Classic Platform
	System Template	
	Specification of ECU Configuration Parameters	
	Requirements on Communication Management	Adaptive Platform
	Specification of Communication Management	
	Specification of Manifest	
	Requirements on Manifest Specification	
Non-Volatile Data Handling Enhancements	List of Basic Software Modules	Classic Platform
	Software Component Template	
	Specification of ECU Configuration Parameters (XML)	
	Specification of NVRAM Manager	
	Specification of Flash EEPROM Emulation	
	Specification of Diagnostic Communication Manager	
	Requirements on Memory Services	
	Specification of RTE Software	
	Specification of Bulk NvData Manager	
	Layered Software Architecture	
	Basic Software UML Model	
	Requirements on Diagnostics	Foundation
FirmwareoverTheAir	**Requirements on Firmware Over-The-Air**	Classic Platform
	Explanation of Firmware Over-The-Air	

Table 2.1: Impact of Concepts

2.1.2 Specifications

2.1.2.1 New Specifications

The following new specifications were introduced via concepts:

- Specification of Basic Software Multicore Library (UID 946, SWS)
- Requirements on Firmware Over-The-Air (UID 944, RS)
- Explanation of Firmware Over-The-Air (UID 945, EXP)
- Specification of Bulk NvData Manager (UID 949, SWS)

In addition to the above listed new specifications, the following document was added with R19-11:

- Explanatory Document for usage of AUTOSAR RunTimeInterface (UID 896, EXP)

2.1.2.2 Migrated Specifications

With this release, the following specifications were moved from AUTOSAR Classic Platform to the AUTOSAR Foundation standard:

- Requirements on E2E Communication Protection (UID 651, SRS) to Requirements on E2E (UID 847, RS)

2.1.2.3 Obsolete Specifications

The following specifications are set to status "obsolete" in this release:

- Application Interfaces User Guide (UID 442, EXP)
- General Requirements on Methodology and Templates (UID 604, RS)
- Requirements on AUTOSAR Features (UID 294, RS)

2.1.2.4 Removed specifications

The following specifications where set to status "removed" in this release and hence are not released anymore:

- Specification of LIN Network Management (UID 297, SWS)
- Requirements on E2E Communication Protection (UID 651, SRS)
- Requirements on Interaction with Behavioral Models (UID 102, RS)
- Interaction with Behavioral Models (UID 205, TR)
- Requirements on Interoperability of AUTOSAR Tools (UID 101, RS)
- Interoperability of AUTOSAR Tools (UID 204, TR)

2.1.2.5 Reworked specifications

The following documents have been changed fundamentally in R19-11:

- none

2.1.2.6 Moved specification parts

The following specification parts have been moved to other documents in R19-11.

2.1.2.6.1 Specification parts moved to SWS Key Manager

- Sources
  - Specification of Crypto Service Manager (UID 402, SWS)
  - Specification of Crypto Interface (UID 806, SWS)
  - Specification of Crypto Driver (UID 807, SWS)
- Target
  - Specification of Key Manager (UID 907, SWS)

Old Document	Old Specification Item ID	New Document	New Specification Item ID	Feature name
SWS CryptoService-Manager	SWS_Csm_01022	SWS KeyManager	ECUC_KeyM_00038	Key elements
SWS CryptoService-Manager	SWS_Csm_00953	SWS KeyManager	ECUC_KeyM_00028	Key formats
SWS CryptoService-Manager	SWS_Csm_01031	SWS KeyManager	SWS_KeyM_00056, SWS_KeyM_00057, SWS_KeyM_00059, SWS_KeyM_00060, SWS_KeyM_00061	Certificate service info
SWS CryptoService-Manager	SWS_Csm_91033, SWS_Csm_01036, SWS_Csm_01037	SWS KeyManager	SWS_KeyM_00056, SWS_KeyM_00057	Certificate parsing
SWS CryptoService-Manager	SWS_Csm_01038, SWS_Csm_01040, SWS_Csm_91034	SWS KeyManager	SWS_KeyM_00057, SWS_KeyM_00059, SWS_KeyM_00060, SWS_KeyM_00061	Certificate verifying
SWS CryptoService-Manager	SWS_Csm_01905, SWS_Csm_91041, SWS_Csm_91042	SWS KeyManager	SWS_KeyM_00082	CSI certificate handling
SWS CryptoInterface	SWS_CryIf_00133, SWS_CryIf_00134, SWS_CryIf_91012, SWS_CryIf_00098, SWS_CryIf_00099, SWS_CryIf_00104	SWS KeyManager	SWS_KeyM_00056, SWS_KeyM_00057	Certificate parsing
SWS CryptoInterface	SWS_CryIf_00133, SWS_CryIf_00134, SWS_CryIf_00135, SWS_CryIf_91017, SWS_CryIf_00123, ▽	SWS KeyManager	SWS_KeyM_00057, SWS_KeyM_00059, SWS_KeyM_00060, SWS_KeyM_00061	Certificate verifying

▽

△

Old Document	Old Specification Item ID	New Document	New Specification Item ID	Feature name
	△ SWS_CryIf_00124, SWS_CryIf_00125, SWS_CryIf_00126, SWS_CryIf_00127, SWS_CryIf_00128			
SWS CryptoDriver	SWS_Crypto_00201, SWS_Crypto_00202, SWS_Crypto_00073, SWS_Crypto_91011, SWS_Crypto_00168, SWS_Crypto_00169, SWS_Crypto_00170	SWS KeyManager	SWS_KeyM_00056, SWS_KeyM_00057	Certificate parsing
SWS CryptoDriver	SWS_Crypto_00201, SWS_Crypto_00202, SWS_Crypto_00073, SWS_Crypto_00171, SWS_Crypto_00172, SWS_Crypto_00173, SWS_Crypto_00174, SWS_Crypto_00175, SWS_Crypto_00176, SWS_Crypto_00177, SWS_Crypto_00178	SWS KeyManager	SWS_KeyM_00057, SWS_KeyM_00059, SWS_KeyM_00060, SWS_KeyM_00061	Certificate verifying
SWS CryptoDriver	ECUC_Crypto_00041	SWS KeyManager	ECUC_KeyM_00028	Key formats

2.1.2.6.2 Specification parts moved to RS E2E

- Source: Requirements on E2E Communication Protection (UID 651, SRS)
- Target: Requirements on E2E (UID 847, RS)

SRS_E2E Requirement ID	RS_E2E Requirment ID
SRS_E2E_08540	RS_E2E_08540
SRS_E2E_08538	RS_E2E_08538
SRS_E2E_08528	RS_E2E_08528
SRS_E2E_08527	RS_E2E_08527
SRS_E2E_08529	RS_E2E_08529
SRS_E2E_08530	RS_E2E_08530
SRS_E2E_08531	RS_E2E_08531
SRS_E2E_08533	RS_E2E_08533
SRS_E2E_08534	RS_E2E_08534
SRS_E2E_08536	RS_E2E_08536
SRS_E2E_08537	RS_E2E_08537
SRS_E2E_08539	RS_E2E_08539

2.1.3 Release Documentation

There were no major changes regarding the Release Documentation.

2.2 History information in AUTOSAR

The following diagram shows the location of documentation of changes.

The Change Documentation will be available for Adaptive Platform starting with R20-11.

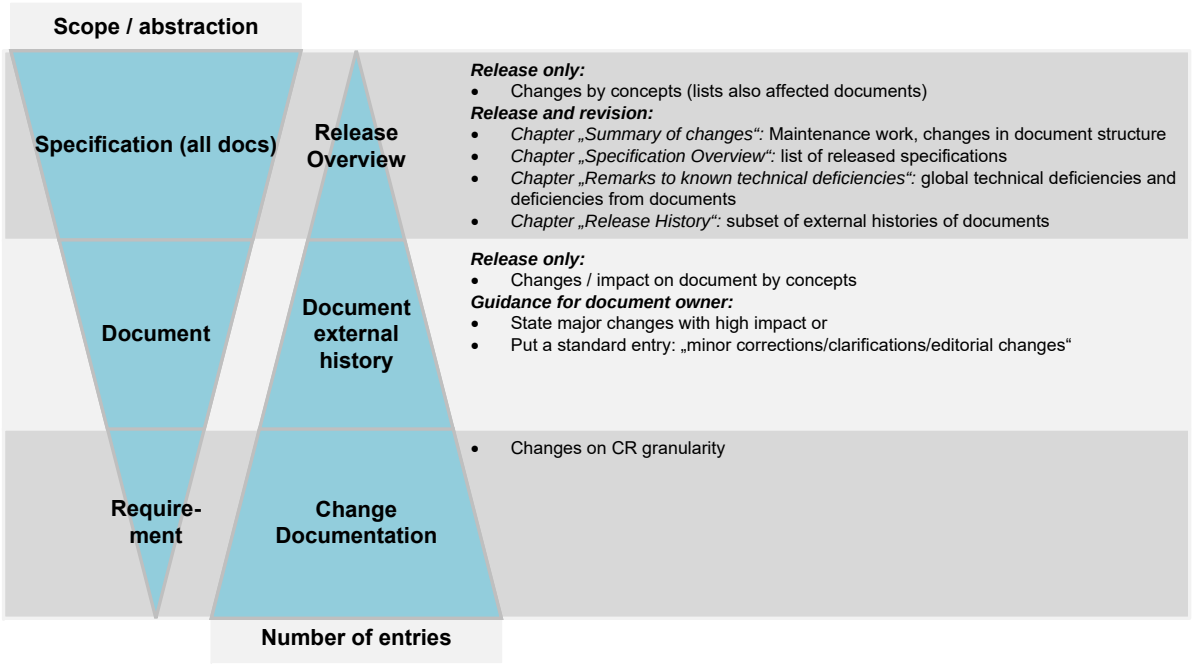

Figure 2.1: History information in AUTOSAR

3 Specification overview

The published specifications are divided into the clusters

- Release Documentation
- Communication
- Memory
- System Services
- MCAL
- IO
- Libraries
- Diagnostics
- Safety
- BSW General
- General
- Methodology and Templates
- Mode Management
- RTE
- Application Interfaces
- Crypto
- Global Time
- SWArch

The assignment of the specifications to these clusters is shown below.

Long Name	File Name	Life cycle changes
Release Documentation		
Classic Platform Release Overview	AUTOSAR_TR_ClassicPlatform ReleaseOverview	
AUTOSAR Classic Platform Specification Hashes	AUTOSAR_TR_ClassicPlatform SpecificationHashes	
Communication		
General Specification on Transformers	AUTOSAR_ASWS_Transformer General	
Requirements on BSW Modules for SAE J1939	AUTOSAR_SRS_SAEJ1939	
Requirements on Bus Mirroring	AUTOSAR_SRS_BusMirroring	
Requirements on CAN	AUTOSAR_SRS_CAN	
Requirements on Communication	AUTOSAR_SRS_COM	
Requirements on Ethernet Support in AUTOSAR	AUTOSAR_SRS_Ethernet	
Requirements on FlexRay	AUTOSAR_SRS_FlexRay	
Requirements on Gateway	AUTOSAR_SRS_Gateway	
Requirements on I-PDU Multiplexer	AUTOSAR_SRS_IPDUMultiplexer	
Requirements on LIN	AUTOSAR_SRS_LIN	
Requirements on Module XCP	AUTOSAR_SRS_XCP	
Requirements on Network Management	AUTOSAR_SRS_Network Management	
Requirements on Secure Onboard Communication	AUTOSAR_SRS_SecureOnboard Communication	
Requirements on SPI Handler/Driver	AUTOSAR_SRS_SPIHandlerDriver	
Requirements on Transformer	AUTOSAR_SRS_Transformer	
Requirements on TTCAN	AUTOSAR_SRS_TTCAN	
Requirements on Vehicle-2-X Communication	AUTOSAR_SRS_V2XCommunication	
Specification of Large Data COM	AUTOSAR_SWS_LargeDataCOM	
Specification of a Request Manager for SAE J1939	AUTOSAR_SWS_SAEJ1939Request Manager	
Specification of a Transport Layer for SAE J1939	AUTOSAR_SWS_SAEJ1939Transport Layer	
Specification of Bus Mirroring	AUTOSAR_SWS_BusMirroring	
Specification of CAN Driver	AUTOSAR_SWS_CANDriver	
Specification of CAN Interface	AUTOSAR_SWS_CANInterface	
Specification of CAN Network Management	AUTOSAR_SWS_CANNetwork Management	
Specification of CAN State Manager	AUTOSAR_SWS_CANStateManager	
Specification of CAN Transceiver Driver	AUTOSAR_SWS_CANTransceiver Driver	
Specification of CAN Transport Layer	AUTOSAR_SWS_CANTransportLayer	
Specification of COM Based Transformer	AUTOSAR_SWS_COMBased Transformer	
Specification of Communication	AUTOSAR_SWS_COM	
Specification of Diagnostic Log and Trace	AUTOSAR_SWS_DiagnosticLogAnd Trace	

△

Long Name	File Name	Life cycle changes
Specification of Diagnostic over IP	AUTOSAR_SWS_DiagnosticOverIP	
Specification of Ethernet Driver	AUTOSAR_SWS_EthernetDriver	
Specification of Ethernet Interface	AUTOSAR_SWS_EthernetInterface	
Specification of Ethernet State Manager	AUTOSAR_SWS_EthernetState Manager	
Specification of Ethernet Switch Driver	AUTOSAR_SWS_EthernetSwitch Driver	
Specification of Ethernet Transceiver Driver	AUTOSAR_SWS_EthernetTransceiver Driver	
Specification of FlexRay AUTOSAR Transport Layer	AUTOSAR_SWS_FlexRayARTransport Layer	
Specification of FlexRay Driver	AUTOSAR_SWS_FlexRayDriver	
Specification of FlexRay Interface	AUTOSAR_SWS_FlexRayInterface	
Specification of FlexRay ISO Transport Layer	AUTOSAR_SWS_FlexRayISO TransportLayer	
Specification of FlexRay Network Management	AUTOSAR_SWS_FlexRayNetwork Management	
Specification of FlexRay State Manager	AUTOSAR_SWS_FlexRayState Manager	
Specification of FlexRay Transceiver Driver	AUTOSAR_SWS_FlexRayTransceiver Driver	
Specification of I-PDU Multiplexer	AUTOSAR_SWS_IPDUMultiplexer	
Specification of LIN Driver	AUTOSAR_SWS_LINDriver	
Specification of LIN Interface	AUTOSAR_SWS_LINInterface	
Specification of LIN State Manager	AUTOSAR_SWS_LINStateManager	
Specification of LIN Transceiver Driver	AUTOSAR_SWS_LINTransceiver Driver	
Specification of Module E2E Transformer	AUTOSAR_SWS_E2ETransformer	
Specification of Module XCP	AUTOSAR_SWS_XCP	
Specification of Network Management for SAE J1939	AUTOSAR_SWS_SAEJ1939Network Management	
Specification of Network Management Interface	AUTOSAR_SWS_Network ManagementInterface	
Specification of PDU Router	AUTOSAR_SWS_PDURouter	
Specification of Secure Onboard Communication	AUTOSAR_SWS_SecureOnboard Communication	
Specification of Service Discovery	AUTOSAR_SWS_ServiceDiscovery	
Specification of Socket Adaptor	AUTOSAR_SWS_SocketAdaptor	
Specification of SOME/IP Transformer	AUTOSAR_SWS_SOMEIPTransformer	
Specification of SPI Handler/Driver	AUTOSAR_SWS_SPIHandlerDriver	
Specification of TCP/IP Stack	AUTOSAR_SWS_TcpIp	
Specification of TTCAN Driver	AUTOSAR_SWS_TTCANDriver	
Specification of TTCAN Interface	AUTOSAR_SWS_TTCANInterface	
Specification of UDP Network Management	AUTOSAR_SWS_UDPNetwork Management	
Specification of Vehicle-2-X Basic Transport	AUTOSAR_SWS_V2XBasicTransport	

▽

△

Long Name	File Name	Life cycle changes
Specification of Vehicle-2-X Facilities	AUTOSAR_SWS_V2XFacilities	
Specification of Vehicle-2-X Geo Networking	AUTOSAR_SWS_V2XGeoNetworking	
Specification of Vehicle-2-X Management	AUTOSAR_SWS_V2XManagement	
Specification of Wireless Ethernet Driver	AUTOSAR_SWS_WirelessEthernet Driver	
Specification of Wireless Ethernet Transceiver Driver	AUTOSAR_SWS_WirelessEthernet TransceiverDriver	
Specification on SOME/IP Transport Protocol	AUTOSAR_SWS_SOMEIPTransport Protocol	
Memory		
Explanation of Firmware Over-The-Air	AUTOSAR_EXP_FirmwareOverTheAir	Initial release
NV Data Handling Guideline	AUTOSAR_EXP_NVDataHandling	
Requirements on EEPROM Driver	AUTOSAR_SRS_EEPROMDriver	
Requirements on Firmware Over-The-Air	AUTOSAR_RS_FirmwareOverTheAir	Initial release
Requirements on Flash Driver	AUTOSAR_SRS_FlashDriver	
Requirements on Flash Test	AUTOSAR_SRS_FlashTest	
Requirements on Memory Hardware Abstraction Layer	AUTOSAR_SRS_Memory HWAbstractionLayer	
Requirements on RAM Test	AUTOSAR_SRS_RAMTest	
Specification of EEPROM Abstraction	AUTOSAR_SWS_EEPROM Abstraction	
Specification of EEPROM Driver	AUTOSAR_SWS_EEPROMDriver	
Specification of Flash Driver	AUTOSAR_SWS_FlashDriver	
Specification of Flash EEPROM Emulation	AUTOSAR_SWS_FlashEEPROM Emulation	
Specification of Flash Test	AUTOSAR_SWS_FlashTest	
Specification of Memory Abstraction Interface	AUTOSAR_SWS_MemoryAbstraction Interface	
Specification of Memory Mapping	AUTOSAR_SWS_MemoryMapping	
Specification of NVRAM Manager	AUTOSAR_SWS_NVRAMManager	
Specification of RAM Test	AUTOSAR_SWS_RAMTest	
System Services		
Recommended Methods and Practices for Timing Analysis and Design within the AUTOSAR Development Process	AUTOSAR_TR_TimingAnalysis	
Requirements on Free Running Timer	AUTOSAR_SRS_FreeRunningTimer	
Requirements on Function Inhibition Manager	AUTOSAR_SRS_FunctionInhibition Manager	
Requirements on Hardware Test Manager on start up and shutdown	AUTOSAR_SRS_HWTestManager	
Requirements on Operating System	AUTOSAR_SRS_OS	
Requirements on Time Service	AUTOSAR_SRS_TimeService	
Specification and Integration of Hardware Test Management at start up and shutdown	AUTOSAR_TR_HWTestManagement IntegrationGuide	

▽

△

Long Name	File Name	Life cycle changes
Specification of Communication Manager	AUTOSAR_SWS_COMManager	
Specification of Default Error Tracer	AUTOSAR_SWS_DefaultErrorTracer	
Specification of Function Inhibition Manager	AUTOSAR_SWS_FunctionInhibition Manager	
Specification of Hardware Test Manager on start up and shutdown	AUTOSAR_SWS_HWTestManager	
Specification of Operating System	AUTOSAR_SWS_OS	
Specification of Time Service	AUTOSAR_SWS_TimeService	
MCAL		
General Requirements on SPAL	AUTOSAR_SRS_SPALGeneral	
Requirements on Core Test	AUTOSAR_SRS_CoreTest	
Requirements on GPT Driver	AUTOSAR_SRS_GPTDriver	
Requirements on MCU Driver	AUTOSAR_SRS_MCUDriver	
Specification of Core Test	AUTOSAR_SWS_CoreTest	
Specification of GPT Driver	AUTOSAR_SWS_GPTDriver	
Specification of MCU Driver	AUTOSAR_SWS_MCUDriver	
IO		
Requirements on ADC Driver	AUTOSAR_SRS_ADCDriver	
Requirements on DIO Driver	AUTOSAR_SRS_DIODriver	
Requirements on I/O Hardware Abstraction	AUTOSAR_SRS_IOHWAbstraction	
Requirements on ICU Driver	AUTOSAR_SRS_ICUDriver	
Requirements on OCU Driver	AUTOSAR_SRS_OCUDriver	
Requirements on Port Driver	AUTOSAR_SRS_PortDriver	
Requirements on PWM Driver	AUTOSAR_SRS_PWMDriver	
Specification of ADC Driver	AUTOSAR_SWS_ADCDriver	
Specification of DIO Driver	AUTOSAR_SWS_DIODriver	
Specification of I/O Hardware Abstraction	AUTOSAR_SWS_IOHardware Abstraction	
Specification of ICU Driver	AUTOSAR_SWS_ICUDriver	
Specification of OCU Driver	AUTOSAR_SWS_OCUDriver	
Specification of Port Driver	AUTOSAR_SWS_PortDriver	
Specification of PWM Driver	AUTOSAR_SWS_PWMDriver	
Libraries		
Macro Encapsulation of Library Calls	AUTOSAR_EXP_MacroEncapsulation ofInterpolationCalls	
Requirements on Libraries	AUTOSAR_SRS_Libraries	
Specification of Basic Software Multicore Library	AUTOSAR_SWS_BSWMulticore Library	
Specification of Bit Handling Routines	AUTOSAR_SWS_BFXLibrary	
Specification of CRC Routines	AUTOSAR_SWS_CRCLibrary	
Specification of Extended Fixed Point Routines	AUTOSAR_SWS_EFXLibrary	
Specification of Fixed Point Interpolation Routines	AUTOSAR_SWS_IFXLibrary	

▽

△

Long Name	File Name	Life cycle changes
Specification of Fixed Point Math Routines	AUTOSAR_SWS_MFXLibrary	
Specification of Floating Point Interpolation Routines	AUTOSAR_SWS_IFLLibrary	
Specification of Floating Point Math Routines	AUTOSAR_SWS_MFLLibrary	
Specification of SW-C End-to-End Communication Protection Library	AUTOSAR_SWS_E2ELibrary	
Diagnostics		
Specification of a Diagnostic Communication Manager for SAE J1939	AUTOSAR_SWS_SAEJ1939 DiagnosticCommunicationManager	
Specification of Diagnostic Communication Manager	AUTOSAR_SWS_Diagnostic CommunicationManager	
Specification of Diagnostic Event Manager	AUTOSAR_SWS_DiagnosticEvent Manager	
Safety		
Overview of Functional Safety Measures in AUTOSAR	AUTOSAR_EXP_FunctionalSafety Measures	
Requirements on Safety Extensions	AUTOSAR_RS_SafetyExtensions	
Requirements on Watchdog Driver	AUTOSAR_SRS_WatchdogDriver	
Safety Use Case Example	AUTOSAR_EXP_SafetyUseCase	
Specification of Watchdog Driver	AUTOSAR_SWS_WatchdogDriver	
Specification of Watchdog Interface	AUTOSAR_SWS_WatchdogInterface	
Specification of Watchdog Manager	AUTOSAR_SWS_WatchdogManager	
Specifications of Safety Extensions	AUTOSAR_TPS_SafetyExtensions	
BSW General		
Basic Software UML Model	AUTOSAR_MOD_BSWUMLModel	
Complex Driver design and integration guideline	AUTOSAR_EXP_CDDDesignAnd IntegrationGuideline	
Description of the AUTOSAR standard errors	AUTOSAR_EXP_ErrorDescription	
Explanation of Error Handling on Application Level	AUTOSAR_EXP_ApplicationLevelError Handling	
Explanation of Interrupt Handling within AUTOSAR	AUTOSAR_EXP_InterruptHandling Explanation	
General Requirements on Basic Software Modules	AUTOSAR_SRS_BSWGeneral	
General Specification of Basic Software Modules	AUTOSAR_SWS_BSWGeneral	
Guide to BSW Distribution	AUTOSAR_EXP_BSWDistribution Guide	
List of Basic Software Modules	AUTOSAR_TR_BSWModuleList	
Modeling Guidelines of Basic Software EA UML Model	AUTOSAR_TR_BSWUMLModel ModelingGuide	
Specification of Communication Stack Types	AUTOSAR_SWS_Communication StackTypes	
Specification of Compiler Abstraction	AUTOSAR_SWS_CompilerAbstraction	
Specification of Platform Types	AUTOSAR_SWS_PlatformTypes	

▽

△

Long Name	File Name	Life cycle changes
Specification of Standard Types	AUTOSAR_SWS_StandardTypes	
General		
Layered Software Architecture	AUTOSAR_EXP_LayeredSoftware Architecture	
Predefined Names in AUTOSAR	AUTOSAR_TR_PredefinedNames	
Requirements on AUTOSAR Features	AUTOSAR_RS_Features	obsolete
Specification of Bulk NvData Manager	AUTOSAR_SWS_BulkNvDataManager	Initial release
Virtual Functional Bus	AUTOSAR_EXP_VFB	
Methodology and Templates		
ARXML Serialization Rules	AUTOSAR_TPS_ARXMLSerialization Rules	
AUTOSAR Feature Model Exchange Format Requirements	AUTOSAR_RS_FeatureModel ExchangeFormat	
AUTOSAR Feature Model Exchange Format	AUTOSAR_TPS_FeatureModel ExchangeFormat	
AUTOSAR Miscellaneous Support Files	AUTOSAR_MOD_MiscSupport	
Basic Software Module Description Template	AUTOSAR_TPS_BSWModule DescriptionTemplate	
Collection of blueprints for AUTOSAR M1 models	AUTOSAR_MOD_GeneralBlueprints	
Collection of constraints on AUTOSAR M1 models	AUTOSAR_TR_AutosarModel Constraints	
Diagnostic Extract Template	AUTOSAR_TPS_DiagnosticExtract Template	
General Requirements on Methodology and Templates	AUTOSAR_RS_MethodologyAnd TemplatesGeneral	obsolete
Generic Structure Template	AUTOSAR_TPS_GenericStructure Template	
Integration of Franca IDL Software Component Descriptions	AUTOSAR_TR_FrancaIntegration	
Interoperability Of Autosar Tools Supplement	AUTOSAR_TR_InteroperabilityOf AutosarToolsSupplement	
Meta Model	AUTOSAR_MMOD_MetaModel	
Meta Model-generated XML Schema	AUTOSAR_MMOD_XMLSchema	
Methodology	AUTOSAR_TR_Methodology	
Modeling Show Cases Examples	AUTOSAR_EXP_ModelingShowCases	
Modeling Show Cases Report	AUTOSAR_TR_ModelingShowCases	
Requirements on Basic Software Module Description Template	AUTOSAR_RS_BSWModule DescriptionTemplate	
Requirements on Diagnostic Extract Template	AUTOSAR_RS_DiagnosticExtract Template	
Requirements on ECU Configuration	AUTOSAR_RS_ECUConfiguration	
Requirements on ECU Resource Template	AUTOSAR_RS_ECUResource Template	
Requirements on Software Component Template	AUTOSAR_RS_SoftwareComponent Template	
Requirements on Standardization Template	AUTOSAR_RS_Standardization Template	

▽

△

Long Name	File Name	Life cycle changes
Requirements on System Template	AUTOSAR_RS_SystemTemplate	
Requirements on Timing Extensions	AUTOSAR_RS_TimingExtensions	
Software Component Template	AUTOSAR_TPS_SoftwareComponent Template	
Specification of ECU Configuration	AUTOSAR_TPS_ECUConfiguration	
Specification of ECU Configuration Parameters (XML)	AUTOSAR_MOD_ECUConfiguration Parameters	
Specification of ECU Resource Template	AUTOSAR_TPS_ECUResource Template	
Specification of Timing Extensions	AUTOSAR_TPS_TimingExtensions	
Standardization Template	AUTOSAR_TPS_Standardization Template	
Standardized M1 Models used for the Definition of AUTOSAR	AUTOSAR_MOD_GeneralDefinitions	
Supplementary material of general blueprints for AUTOSAR	AUTOSAR_TR_GeneralBlueprints Supplement	
Supplementary material of the AUTOSAR XML Schema	AUTOSAR_TR_XMLSchema Supplement	
System Template	AUTOSAR_TPS_SystemTemplate	
XML Schema Production Rules	AUTOSAR_TPS_XMLSchema ProductionRules	
Mode Management		
Guide to Mode Management	AUTOSAR_EXP_ModeManagement Guide	
Requirements on Mode Management	AUTOSAR_SRS_ModeManagement	
Specification of Basic Software Mode Manager	AUTOSAR_SWS_BSWModeManager	
Specification of ECU State Manager	AUTOSAR_SWS_ECUStateManager	
RTE		
Requirements on Runtime Environment	AUTOSAR_SRS_RTE	
Specification of RTE Software	AUTOSAR_SWS_RTE	
Application Interfaces		
Application Design Patterns Catalogue	AUTOSAR_TR_AIDesignPatterns Catalogue	
Application Interface Examples	AUTOSAR_MOD_AISpecification Examples	
Application Interfaces User Guide	AUTOSAR_EXP_AIUserGuide	obsolete
Explanation of Application Interfaces of Occupant and Pedestrian Safety Systems Domain	AUTOSAR_EXP_AIOccupantAnd PedestrianSafety	
Explanation of Application Interfaces of the Body and Comfort Domain	AUTOSAR_EXP_AIBodyAndComfort	
Explanation of Application Interfaces of the Chassis Domain	AUTOSAR_EXP_AIChassis	
Explanation of Application Interfaces of the HMI, Multimedia and Telematics Domain	AUTOSAR_EXP_AIHMIMultimediaAnd Telematics	
Explanation of Application Interfaces of the Powertrain Engine Domain	AUTOSAR_EXP_AIPowertrain	

▽

△

Long Name	File Name	Life cycle changes
Requirements on SW-C and System Modeling	AUTOSAR_RS_SWCModeling	
SW-C and System Modeling Guide	AUTOSAR_TR_SWCModelingGuide	
Unique Names for Documentation, Measurement and Calibration: Modeling and Naming Aspects including Automatic Generation	AUTOSAR_TR_AIMeasurement CalibrationDiagnostics	
XML Specification of Application Interfaces	AUTOSAR_MOD_AISpecification	
Crypto		
Requirements on Crypto Stack	AUTOSAR_SRS_CryptoStack	
Specification of Crypto Driver	AUTOSAR_SWS_CryptoDriver	
Specification of Crypto Interface	AUTOSAR_SWS_CryptoInterface	
Specification of Crypto Service Manager	AUTOSAR_SWS_CryptoService Manager	
Specification of Key Manager	AUTOSAR_SWS_KeyManager	
Utilization of Crypto Services	AUTOSAR_EXP_UtilizationOfCrypto Services	
Global Time		
Specification of Synchronized Time-Base Manager	AUTOSAR_SWS_SynchronizedTime BaseManager	
Specification of Time Synchronization over CAN	AUTOSAR_SWS_TimeSyncOverCAN	
Specification of Time Synchronization over Ethernet	AUTOSAR_SWS_TimeSyncOver Ethernet	
Specification of Time Synchronization over FlexRay	AUTOSAR_SWS_TimeSyncOverFlex Ray	
SWArch		
Explanatory Document for usage of AUTOSAR RunTimeInterface	AUTOSAR_EXP_ClassicPlatformARTI	Initial release
Requirements on Debugging, Tracing and Profiling support of AUTOSAR Components	AUTOSAR_RS_ClassicPlatformDebug TraceProfile	
Specification of AUTOSAR Run-Time Interface	AUTOSAR_SWS_ClassicPlatformARTI	

Table 3.1: Specification Overview

4 Remarks to known technical deficiencies

The technical deficiencies per specification are - if applicable - mentioned inside the respective specification in a chapter "Known Limitations" located after the table of contents.

The following technical deficiencies are to be mentioned:

- **Explanation of Firmware Over-The-Air (UID 945, EXP)**
  This Explanatory is set to status "draft" in R19-11.
  Use cases and topics that are not covered within this document:
  - Update of ECUs without UDS support
  - Verification strategy of the newly flashed SW
  - ECU SW version handling and checking (vendor specific)
  - ECU SW compatibility/integrity check (vendor specific)
  - Any details about the SW architecture of the memory stack
- **Specification of TCP/IP Stack (UID 617, SWS)**
  This document does not cover the assignment of UDP or TCP port numbers. There is no reserved space within the IANA assigned number range. Each implementer is responsible for managing the used port numbers.

  This document does not cover the management of IP addresses. This might be done dynamically, e.g. by using DHCP, or statically. It is the implementer's responsibility to prevent address conflicts and achieve compliance with IANA address assignments.

  This specification does not prescribe a certain physical layer or data rate.

  Although a CDD interface is specified, allowing additional upper layer modules, a fan-out of one socket to multiple upper layer modules is not intended to be supported.

  The AUTOSAR TLS implementation has the following limitations:
  - A TLS implementation shall not support data compression or decompression.
  - Session renegotiation shall not be supported.
  - No support for secure connection over UDP (e.g. for DTLS)
  - No support of FQDN
  - No client Hello padding extension IETF RFC7685
  - No session hash and extended master secret IETF RFC 7627
  - No support for TLS versions lower than 1.2.

  - No support for dynamic "downgrading" of a TCP connection with an established TLS connection to a plain TCP connection (without TLS)
  - Static TLS connection assignment is bound to the port configuration of the server. Thus, using different TLS settings for different connections (possibly originating from different clients) to the same server port is not possible.

  Please be aware that all specification items related to TLS are marked as 'DRAFT', as their verification is still pending and might be subject to change within the next releases.

  The AUTOSAR IPsec implementation has the following limitations:

  - IPsec in "tunnel mode" is not supported right now. Transport mode only.
  - IPv6 is not supported
  - Multicast is not supported

- **Specification of Socket Adaptor (UID 416, SWS)**
  The transmission of data using TCP/IP over Ethernet requires about 60 bytes of header information. This implies that for small messages the header overhead may reach an unacceptably high percentage.

  To avoid further protocol overhead, the use of a single socket connection per PDU is described here. However, this solution is very resource intensive, particularly if many small PDUs are to be transmitted. One solution described here as an option is to add a small PDU header, containing an ID and length information. This enables transmission of multiple PDUs via one socket connection. Additionaly a resource conservation scheme is included in this specification as an option.

  This document does not cover the assignment of UDP or TCP port numbers. There is no reserved space within the IANA assigned number range. Each implementer is responsible for managing the used port numbers.

  This document does not cover the management of IP addresses. This might be done dynamically, e.g. by using DHCP, or statically. It is the implementers responsibility to prevent address conflicts and achieve compliance with IANA address assignments.

  This specification does not prescribe a certain physical layer or data rate.

- **Recommended Methods and Practices for Timing Analysis and Design within the AUTOSAR Development Process (UID 645, TR)**
  Note that subsection 8.1.1 AUTOSAR Classic Platform Operating System contains content that was part of CONC_628. As CONC_628 was not validated for the AUTOSAR Classic Platform release 4.4.0, this content is added as draft to the current AUTOSAR release.

- **Specification of CAN Network Management (UID 13, SWS)**
  One channel of CanNm is associated with only one network management cluster in one network.

One network management cluster can have only one channel of CanNm in one node.

One channel of CanNm is associated with only one network within the same ECU.

CanNm is only applicable for CAN1 systems.

- **Specification of FlexRay State Manager (UID 254, SWS)**
  This specification only defines the straightforward case for starting and stopping the communication on a FlexRay cluster.

  For the case of multiple CC of one ECU assigned to one FlexRay cluster some items are left open for the implementation:

  – Which CC is used to transmit the wakeup pattern

  – Handling of inconsistent POC states in the CCs

- **Specification of Network Management for SAE J1939 (UID 612, SWS)**
  The J1939 Network Management module does not support all features defined in [1], especially:

  – Changing the address of a node after reception of CommandedAddress or after an address loss.

  – Changing the NAME of a node using the Name Management protocol.

  – Detection of address violations by messages other than AddressClaimed.

- **Specification of a Request Manager for SAE J1939 (UID 611, SWS)**
  The J1939 Request Manager only implements Request, Request2, and Acknowledgement PGs. It does not provide support for the Transfer PG.

- **Specification of a Transport Layer for SAE J1939 (UID 425, SWS)**
  The AUTOSAR architecture contains several communication system specific transport layers (J1939Tp, CanTp, FrTp, etc.). All of these modules need to have identical APIs, with the exception of API functions for which the PduR has separate configuration abilities.

  The J1939Tp module does not implement the TriggerTransmit API, because it is only needed for time triggered bus architectures.

- **Specification of Flash EEPROM Emulation (UID 286, SWS)**
  The synchronization of a potential parallel access (e.g. BulkNvDataManager) to the underlying flash driver is not part of this AUTOSAR release.

- **Specification of Core Test (UID 259, SWS)**
  A Core test module implementation might be limited to be executed during power-up/start-up time where core resources are not shared among different active AUTOSAR related software tasks or hardware-entities (e.g. IRQ-controller, DMA, Cache, MMU/MPU and MemoryIF)

  -OR-

might be limited to test resources which are not shared during runtime software execution (e.g. ALU and CPU-registers). This is overall automotive system architecture dependent and cannot be covered in a MCAL Core Test SWS specification.

There must be a managing entity or architecture available who manages tasks like 'hardware-resource-access-managing' due to the inability of a MCAL-driver to handle such tasks on its own.

- **Specification of Ethernet Transceiver Driver (UID 431, SWS)**
  - Ethernet transceiver module handles only single thread execution
  - Execution cannot pre-empt ifself
- **Specification of Module E2E Transformer (UID 650, SWS)**
  E2ETransformer now supports Client-Server Communication, except the following
  - NoNewData checks are only supported for polling-based Client-Server Communication and Sender-Receiver Communication.
  - E2E Profiles 1 and 2 do not support Client-Server Communication at all.
- **Specification of Ethernet Interface (UID 417, SWS)**
  - The Ethernet Interface is conceptually able to access one or more Ethernet Driver and one or more Ethernet Transceiver Driver.
  - It is not possible to transmit data which exceeds the available buffer size of the used Ethernet controller. Longer data has to be transmitted using the Internet Protocol (IP) or Transmission Control Protocol (TCP).
- **Specification of Flash Driver (UID 25, SWS)**
  - The flash driver only erases or programs complete flash sectors respectively flash pages, i.e. it does not offer any kind of re-write strategy since it does not use any internal buffers.
  - The flash driver does not provide mechanisms for providing data integrity (e.g. checksums, redundant storage, etc.).
- **Specification of RAM Test (UID 76, SWS)**
  During the execution of a RAM test algorithm, no other software shall be allowed to modify the RAM area under test.

  In case of background test, the testing code shall be implemented in small atomic pieces in order to accomplish this.

In case of foreground test, it is assumed that the test environment provides the conditions for exclusive access to the tested RAM area.

The rationale behind this requirement is the incapability of the RAM test module to ensure data consistency (e.g. during an NMI, or during a DMA transfer).

The implementer shall provide integration hints for each algorithm, e.g. "do not use in parallel with a DMA".

When testing shared memory in a multi-core system it might not be possible to get exclusive access to more than one memory cell via interrupt locking. In this case, the usage of a test configuration for shared memory blocks must be restricted to foreground tests and to specific ECU states, see 3 Related Documentation and SWS_RamTst_00203 for additional information.

In a multi-core system, disabling the interrupts does not guarantee atomicity for more than a single memory access. Since a RAM test operation consists of more than a single memory access, a more sophisticated mechanism is needed to realize atomicity. Therefore, different solutions for shared and non-shared RAM are required.

- **Specification of a Diagnostic Communication Manager for SAE J1939 (UID 610, SWS)**
  The J1939 Diagnostic Communication Manager implements only a subset of "Diagnostic messages" as defined in Table 1: Supported DMx messages.

  The DM13 does not support "Suspend Signal" "Suspend Duration".

  NACK is not provided for received DMx messages that are not supported or not configured. This restriction mainly affects handling of DM07 and DM13.

- **Specification of Time Synchronization over Ethernet (UID 676, SWS)**

  - No support of BMCA protocol.
  - No support of Announce and Signaling messages.
  - The reception of a Pdelay_Req is not taken as a pre-condition to start with the transmission of Sync messages
  - The Rate Correction will be performed by the StbM, based on Sync messages, which does not require the Pdelay mechanism, though the IEEE Standard mandates to calculate the rate correction based on Pdelay messages. This is considered to be a deviation from the IEEE-Standard, but it is considered to be interoperable. For some applications, e.g. for Audio/Video, it might be necessary to use Pdelay based Rate Correction performed by EthTSyn itself, which is optional and not considered by this specification.
  - The Time Validation use case (Time Validation enabled) requires to perform the Pdelay measurement with timestamps taken from the local instance of

that Global Time that needs to be validated. This is considered to be a deviation from the IEEE-Standard, but it is considered to be interoperable.

  - Time measurement with Switches (Time Aware Bridges) are not supported for the Time Validation use case.
  - EthTSyn will not maintain the Ethernet HW clock, but may use it as a source for the Virtual Local Time.
  - While IEEE 802.1AS states, that IEEE 802.1AS message shall not have a VLAN tag nor a priority tag, EthTSyn would allow Time Synchronization on VLANs under the condition, that the switch HW supports forwarding of reserved multicast address using the range of 01:80:C2:00:00:00 .. 0F.

- **Specification of Synchronized Time-Base Manager (UID 421, SWS)**

  - OS ScheduleTable: The StbM considers only the case when the targeted OS ScheduleTable is explicitly synchronized. The implicit synchronization does not affect the StbM.
  - StbMSynchronizedTimeBaseIdentifier: The StbMSynchronizedTimeBaseIdentifier range (128 .. 65535) is currently reserved and might still be used by legacy applications (implementing Triggered Customers). The ID range will however be reassigned to new features in the next release. Legacy applications will then no longer be supported.
  - Mode switches: The Synchronized Time-Base Manager does not deal with mode switches during runtime.
  - Configuration: Postbuild configuration of the StbM is limited to enabling or disabling the functionality of a system wide Global Time Master for a Time Base
  - Time Validation: Time Validation cannot be used with postbuild configuration, i.e., Time Validation has to be switched off when postbuild configuration of Master/Slave role of the StbM is used.
    For each Time Base with Time Validation enabled, if StbM is configured as Time Gateway for that Time Base, only one Master Domain per Timesync Module can be linked to that Time Base.

- **Specification of Time Synchronization over CAN (UID 674, SWS)**
  The current version of CanTSyn does not support hardware timestamping capabilities.
  The first consequence is that the Time Synchronization is less accurate due to Rx-Tx-ISR latencies and execution time until the Virtual Local Time is retrieved.
  The second consequence is the need of not nested interrupts in the CAN driver for the Global Time PDUs (i.e., it is strongly recommended not to invoke the TX confirmation and RX indication functions in polling mode).

The Time Base in the SYNC and OFS messages is limited to 32 bit, wherefore the maximum supported time value is 4294967295 seconds $(2^{32} - 1)$.

Time Masters, Time Gateways and Time Slaves shall work with a Time Base reference clock with a worst-case accuracy of $2\mu s$.

- **Specification of Time Synchronization over FlexRay (UID 675, SWS)**
  Time Masters, Time Gateways and Time Slaves shall work with a Time Base reference clock with a worst-case accuracy of $2\mu s$.

  The Time Base in the OFS messages is limited to 32 bit, wherefore the maximum supported time value is 4294967295 seconds $(2^{32} - 1)$.

- **Specification of Network Management Interface (UID 228, SWS)**

  - The Generic Network Management Interface can only be applied to communication systems that support broadcast communication and 'bus-sleep mode'.
  - There is only one instance of the Generic Network Management Interface layer for all NM-Clusters. This instance manages all channels where a NM is used.
  - The Generic Network Management Interface shall only include the common modes, definitions and return values of different bus specific NM layers.
  - The Generic Network Management Interface shall only include the common modes, definitions and return values of different bus specific NM layers.

- **Specification of Bus Mirroring (UID 873, SWS)**
  The Bus Mirroring module cannot be used to influence the traffic on one of the buses configured as a source bus. To ensure this and to avoid loop-back of messages leading to bus overload, the generation tool shall ensure that no bus is connected to the Bus Mirroring module both as source and destination bus (see SWS_Mirror_00001).

  The Bus Mirroring module is controlled by a diagnostic control application through the dedicated (service) API listed in chapter 8. The control functionality is made acces- sible to a diagnostic tester by special diagnostic services, which are handled by the DCM and implemented by the diagnostic control application. The DCM provides the necessary security to exclude inadvertent activation of the Bus Mirroring. The Bus Mirroring module does not provide another control interface, and it does not receive control messages on the destination bus.

  In general, the Bus Mirroring module does not support source buses that have a larger frame size or more additional information than the destination bus can carry, e.g. CAN- FD to CAN, CAN to LIN, FlexRay to CAN, Ethernet to CAN, or Ethernet to FlexRay.

  The Bus Mirroring module does not fragment mirrored frames.

The Bus Mirroring module will only mirror traffic that is actually received or transmitted by the bus interface modules. For CAN this means that besides the transmitted frames only those data frames that pass the hardware filter will be mirrored, and that remote frames and error frames will not be mirrored. For LIN, slave-to-slave communication will not be mirrored by a LIN master. And for FlexRay, only transmitted frames and those received frames for which reception buffers are assigned (possibly as a FIFO) will be mirrored.

Another limitation of the mirroring from a FlexRay source bus concerns the reported time stamps and cycles. The Timestamp reported for a FlexRay frame contains the time when the corresponding job list entry was executed. The actual transmission time has to be calculated from the slot ID contained in the reported FrameID. The cycle contained in the reported FrameID is accurate only for received frames and frames transmitted in the static segment. For frames transmitted in the dynamic segment, the reported cycle can be inaccurate because it can happen that a frame cannot be transmitted in the expected cycle, it is then deferred to the next suitable cycle.

A re-serialization of received serialized frames shall not be done by the Bus Mirroring module, because that would require too much resources. Instead, the serialized PDUs shall be routed directly to the destination bus.

The Bus Mirroring module will also not support the forwarding from Ethernet to Ethernet. This use case is already covered by the Port Mirroring feature of the AUTOSAR Ethernet Switch Driver.

- **Specification of Ethernet Driver (UID 430, SWS)**
  The Ethernet Driver module is only able to handle a single thread of execution. The execution must not be pre-empted by itself.

  It is not possible to transmit data which exceeds the available buffer size of the used controller. Longer data has to be transmitted using the Internet Protocol (IP) or Transmission Control Protocol (TCP).

  Depending on the Ethernet hardware, it may become necessary that implementations deviate from API specifications in respect to the asynchronous/synchronous behaviour.

- **Specification of ECU State Manager (UID 78, SWS)**
  ECUs cannot always be switched off (i.e. zero power consumption).

- **Specification of Wireless Ethernet Driver (UID 798, SWS)**

  – It is not possible to transmit data which exceeds the available buffer size of the used controller.

  – Wireless Communication supports IEEE 802.11p only. Other 802.11 standards (e.g. for infrastructure networks and integration with TCP/IP) can be extended in future releases of the AUTOSAR standard.

- The V2X modules follow the guidance regarding the Day-1 scenarios defined by the Basic System Standards Profile from Car-2-Car-Consortium.
- AUTOSAR R19-11 only focuses on the European version of car-to-car communication as defined by ETSI. Extension to other regions are planned for future releases of the AUTOSAR standard.
- The Microcontroller Abstraction Layer Multi-Core Distribution Concept is implemented as "‘draft"’ in this software specification. Refer to chapter 10 for more information.

5 Release history

5.1 Release R19-11

Release R19-11 was originally released on 28 November 2019.

Name	Specification history entry
Application Design Patterns Catalogue	• Subfunctions per layer defined • Capability information introduced • FAQ and known issues section implemented • Separation of Sensor and Actuator in namespace • Changed Document Status from Final to published
Application Interface Examples	• No content changes • Changed Document Status from Final to published
Application Interfaces User Guide	• Editorial changes • Set Document to obsolete, as methodology is implemented in AI-Tool incl. online help
ARXML Serialization Rules	• no content changes • Changed Document Status from Final to published
AUTOSAR Feature Model Exchange Format Requirements	• Editorial changes • Changed Document Status from Final to published
AUTOSAR Feature Model Exchange Format	• Editorial changes • Changed Document Status from Final to published
Basic Software Module Description Template	• added constraint for the use task or cat2Isr • Editorial changes • Changed Document Status from Final to published
Classic Platform Release Overview	• Release Life Cycle Status: R19-11 is in Evolution, R19-11 supersedes R4.4
Collection of constraints on AUTOSAR M1 models	• Updated constraints according to changes in SWS and TPS documents • Changed Document Status from Final to published
Complex Driver design and integration guideline	• Add a note in the 7.3.8 chapter • Changed Document Status from Final to published
Description of the AUTOSAR standard errors	• No content changes • Changed Document Status from Final to published
Diagnostic Extract Template	• Increase support for variant handling • Improved Dem upstream mapping • Support for custom service instances • minor corrections / clarifications /editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published
Explanation of Application Interfaces of Occupant and Pedestrian Safety Systems Domain	• No content changes • Changed Document Status from Final to published

△

Name	Specification history entry
Explanation of Application Interfaces of the Body and Comfort Domain	• Editorial changes • Changed Document Status from Final to published
Explanation of Application Interfaces of the Chassis Domain	• No content changes • Changed Document Status from Final to published
Explanation of Application Interfaces of the HMI, Multimedia and Telematics Domain	• No content changes • Changed Document Status from Final to published
Explanation of Application Interfaces of the Powertrain Engine Domain	• Remove obsolete Reference [2] Table of Application Interfaces and change to [3] • Update Table of Contents • Remove obsolete Reference [2] Table of Application Interfaces and change to [3] • Update Table of Contents • Update of figures to reflect the latest definitions of Transmission - Engine Interfaces • Extension of chapter 3.1 by section "Consideration of the inertia in torque signals" • Chapter 6.3.6: "Special rules for engine-transmission interfaces" modified • Changed Document Status from Final to published
Explanation of Error Handling on Application Level	• No content changes • Changed Document Status from Final to published
Explanation of Firmware Over-The-Air	• Initial release
Explanation of Interrupt Handling within AUTOSAR	• No content changes • Changed Document Status from Final to published
Explanatory Document for usage of AUTOSAR RunTimeInterface	• Initial release
General Requirements on Basic Software Modules	• No content changes • Changed Document Status from Final to published
General Requirements on Methodology and Templates	• Document set to obsolete • Changed Document Status from Final to obsolete
General Requirements on SPAL	• No content changes • Changed Document Status from Final to published
General Specification of Basic Software Modules	• Include guard for header files • minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published
General Specification on Transformers	• Added chapter 8.2.1 Std_TransformerForward • Editorial changes • Changed Document Status from Final to published
Generic Structure Template	• Refine Splitable • Extent AttributeValueVariationPoint • Introduce TracableTable • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Guide to BSW Distribution	• Incorporation of concept "BSW Multicore Distribution" • Changed Document Status from Final to published
Guide to Mode Management	• No content changes • Changed Document Status from Final to published
Integration of Franca IDL Software Component Descriptions	• Editorial changes • Changed Document Status from Final to published
Layered Software Architecture	• Incorporated new concepts for Atomic multicore safe operations, Signal-service-translation, NV data handling enhancement • Changed Document Status from Final to published
List of Basic Software Modules	• Added Bulk NvData Manager • Added BSW Multicore Library • Changed Document Status from Final to published
Macro Encapsulation of Library Calls	• No content changes • Changed Document Status from Final to published
Methodology	• Editorial changes • Handling of Platform/Standard Types as Blueprints • Removed references to TR IOAT • Changed Document Status from Final to published
Modeling Guidelines of Basic Software EA UML Model	• described modeling of Development Errors, Runtime Errors, and Transient Faults • Changed Document Status from Final to published
Modeling Show Cases Report	• Add Show case Structured Requirement • Changed Document Status from Final to published
NV Data Handling Guideline	• No content changes • Changed Document Status from Final to published
Overview of Functional Safety Measures in AUTOSAR	• Removed Duplicated IDs • Changed Document Status from Final to published
Predefined Names in AUTOSAR	• Included abbreviations for ARTI • Removed references to TR_InteroperabilityOfAutosarTools • Changed Document Status from Final to published
Recommended Methods and Practices for Timing Analysis and Design within the AUTOSAR Development Process	• Added section 5.8 on introduction of service oriented communication • Minor updates and improvements • Editorial changes • Changed Document Status from Final to published
Requirements on ADC Driver	• No content changes • Changed Document Status from Final to published
Requirements on AUTOSAR Features	• Added protocol IPsec • Changed Document Status from Final to obsolete
Requirements on Basic Software Module Description Template	• Changed Document Identification No to 86

▽

△

Name	Specification history entry
Requirements on BSW Modules for SAE J1939	• No Content Changes • Changed Document Status from Final to published
Requirements on Bus Mirroring	• Fixed referenced documents • Changed Document Status from Final to published
Requirements on CAN	• Bus-independent solution regarding channel states upon initialization • Changed Document Status from Final to published
Requirements on Communication	• No content changes • Changed Document Status from Final to published
Requirements on Core Test	• No content changes • Changed Document Status from Final to published
Requirements on Crypto Stack	• Updated items in glossary and abbreviation list • Changed Document Status from Final to published
Requirements on Debugging, Tracing and Profiling support of AUTOSAR Components	• No content changes • Changed Document Status from Final to published
Requirements on Diagnostic Extract Template	• No content changes • Changed Document Status from Final to published
Requirements on DIO Driver	• No content changes • Changed Document Status from Final to published
Requirements on ECU Configuration	• Changed Document Identification No to 85 • Changed Document Status from Final to published
Requirements on ECU Resource Template	• Editorial changes
Requirements on EEPROM Driver	• No content changes • Changed Document Status from Final to published
Requirements on Ethernet Support in AUTOSAR	• Introduction of Ipsec • Changed Document Status from Final to published
Requirements on Firmware Over-The-Air	• Initial release
Requirements on Flash Driver	• No content changes • Changed Document Status from Final to published
Requirements on Flash Test	• No content changes • Changed Document Status from Final to published
Requirements on FlexRay	• Modification of inizialization requirments • Changed Document Status from Final to published
Requirements on Free Running Timer	• No content changes • Changed Document Status from Final to published
Requirements on Function Inhibition Manager	• No content changes • changed Document Status from Final to published
Requirements on Gateway	• No content changes • Changed Document Status from Final to published
Requirements on GPT Driver	• No content changes • Changed Document Status from Final to published
Requirements on Hardware Test Manager on start up and shutdown	• No content changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Requirements on I/O Hardware Abstraction	• No content changes • Changed Document Status from Final to published
Requirements on ICU Driver	• No content changes • Changed Document Status from Final to published
Requirements on I-PDU Multiplexer	• No Content Changes • Changed Document Status from Final to published
Requirements on Libraries	• No content changes • Changed Document Status from Final to published
Requirements on LIN	• No Content Changes • Changed Document Status from Final to published
Requirements on MCU Driver	• No content changes • Changed Document Status from Final to published
Requirements on Memory Hardware Abstraction Layer	• No content changes • Changed Document Status from Final to published
Requirements on Memory Services	• Added SRS_Mem_00139 • Changed Document Status from Final to published
Requirements on Mode Management	• No content changes • Changed Document Status from Final to published
Requirements on Module XCP	• No content changes • Changed Document Status from Final to published
Requirements on Network Management	• Remove references to removed requirements • Editorial changes • Changed Document Status from Final to published
Requirements on OCU Driver	• No content changes • Changed Document Status from Final to published
Requirements on Operating System	• No content changes • Changed Document Status from Final to published
Requirements on Port Driver	• No content changes • Changed Document Status from Final to published
Requirements on PWM Driver	• No content changes • Changed Document Status from Final to published
Requirements on RAM Test	• No content changes • Changed Document Status from Final to published
Requirements on Runtime Environment	• RTE Implementation Plug-ins: Set [SRS_Rte_00300] - [SRS_Rte_00317] to type valid • Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding (TLV): Set [SRS_Rte_00261] to valid • Category 2 interrupts and RunnableEntities • Changed Document Status from Final to published
Requirements on Safety Extensions	• No content changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Requirements on Secure Onboard Communication	• Added requirement regarding the description of the complete TP TX communication flow towards the Upper Layer. • Changed Document Status from Final to published
Requirements on Software Component Template	• Changed status of RS_SWCT_03320 to valid. • Changed Document Status from Final to published
Requirements on SPI Handler/Driver	• No Content Changes • Changed Document Status from Final to published
Requirements on Standardization Template	• No content changes • Changed Document Status from Final to published
Requirements on SW-C and System Modeling	• No content changes • Changed Document Status from Final to published
Requirements on System Template	• Added requirement for Signal-To-Service Translation • Changed Document Status from Final to published
Requirements on Time Service	• No content changes • Changed Document Status from Final to published
Requirements on Timing Extensions	• The status of requirement RS_TIMEX_00021 is set to "removed" and all remaining references within the document have been cleaned up. • Rationale has been given and reference to use case has been added for RS_TIMEX_00014. • Changed Document Status from Final to published
Requirements on Transformer	• Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding set to valid. • Editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published
Requirements on TTCAN	• No content changes • Changed Document Status from Final to published
Requirements on Vehicle-2-X Communication	• Major rework of document references according BSP V1.3 • Minor corrections • Changed Document Status from Final to published
Requirements on Watchdog Driver	• No content changes • Changed Document Status from Final to published
Safety Use Case Example	• No content changes • Changed Document Status from Final to published
Software Component Template	• Support for optimized access to coding data • Support for meta-data on application level • Support for optimized storing of bitfields in NvRAM • minor corrections / clarifications /editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification and Integration of Hardware Test Management at start up and shutdown	• No content changes • Changed Document Status from Final to published
Specification of Large Data COM	• Clean up diagrams in chapter 10 • Changed Document Status from Final to published
Specification of RTE Software	• Support for meta-data on application level • Support for direct access to the RamMirror of a NvBlockComponent • Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
Specification of a Diagnostic Communication Manager for SAE J1939	• Added generic DMx support • Fixed tracing to RS Diagnostics • Improved API and service port tables • Changed Document Status from Final to published
Specification of a Request Manager for SAE J1939	• Cleaned up EcuC diagrams • Improved service port tables • Changed Document Status from Final to published
Specification of a Transport Layer for SAE J1939	• No Content Changes • Changed Document Status from Final to published
Specification of ADC Driver	• API changed to asynchronous API: – Adc_SetupResultBuffer, – Adc_EnableHardwareTrigger, – Adc_DisableHardwareTrigger, – Adc_EnableGroupNotification, – Adc_DisableGroupNotification • Changed Document Status from Final to published
Specification of AUTOSAR Run-Time Interface	• Added expression syntax • Corrected trace macros and ARTI class names • Added and extended several configuration parameters • Corrected SWS item references • Changed Document Status from Final to published
Specification of Basic Software Mode Manager	• Introduced action list priority parameter • Added configuration options for enabling/disabling SdServiceGroups • Removal of obsolete APIs: BswM_EcuM_CurrentState and BswM_EcuM_RequestedState • Editorial Changes • Changed Document Status from Final to published
Specification of Basic Software Multicore Library	• Initial release
Specification of Bit Handling Routines	• Editorial changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of Bulk NvData Manager	• Initial release
Specification of Bus Mirroring	• Added multi-partition support DRAFT • Fixed configurable number of PDUs • Reworked requirements to avoid references to sections • Changed Document Status from Final to published
Specification of CAN Driver	• Minor corrections / clarifications / editorial changes; • Changed Document Status from Final to published
Specification of CAN Interface	• Update reference to ISO11898-1:2015 • Minor corrections • Editorial changes • Changed Document Status from Final to published
Specification of CAN Network Management	• Clarification for CAN FD usage • Extended Wait Bus Sleep Handling • Changed Document Status from Final to published
Specification of CAN State Manager	• Fixed Change_Baudrate-Statemachine for NoCom • Added GetPduMode-Interface to list. • Inconsistent behavior due to REPEAT_MAX / No Never-Give-Up Strategy fixed • Changed Document Status from Final to published
Specification of CAN Transceiver Driver	• Sequence diagram De-Initialization (SPI Synchronous) and De-Initialization (SPI Asynchronous) split into different pages. • Minor correction in CanTrcv initialization functionality. • Changed Document Status from Final to published
Specification of CAN Transport Layer	• Added configuration diagrams • Clarifications • Changed Document Status from Final to published
Specification of COM Based Transformer	• editorial • Changed Document Status from Final to published
Specification of Communication	• MetaData handling above RTE (CONC_650) • BSW Multicore Distribution as draft (CONC_643) • minor corrections / clarifications / editorial changes • Changed Document Status from Final to published
Specification of Communication Manager	• Introduce handling of PNC coordinator if serval ComM channels have the same PNC assignment but PncGatewayTypeEnum is set to "none." • Enabled ComM to be used for BSW distribution (multicore use case) • Minor corrections • Changed Document Status from Final to published
Specification of Communication Stack Types	• Renamed of general types headers • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of Compiler Abstraction	• No content changes • Changed Document Status from Final to published
Specification of Core Test	• Incorporated changes to support MCALMulticoreDistribution • Changed Document Status from Final to published
Specification of CRC Routines	• No content changes • Changed Document Status from Final to published
Specification of Crypto Driver	• Minor corrections and editorial changes • Cleanup of return code and DET error • Default RNG configuration for CryptoDriver Objects • Clarifcation on Read/Write access for key element. • Remove certificate support functions • Remove virtual key references • Changed Document Status from Final to published
Specification of Crypto Interface	• Minor changes • Clarify key ID handling • Remove certificate handling • Cleanup of DET and return errors • Changed Document Status from Final to published
Specification of Crypto Service Manager	• Bringing return values of all services and interfaces to one line • added functionality and description of elliptic curves • Callback notification modified • Editorial changes • Changed Document Status from Final to published
Specification of Default Error Tracer	• Editorial changes in Upracing (from "SRS_"' to "RS_") • Changed Document Status from Final to published
Specification of Diagnostic Communication Manager	• Incorporation of Concept NVData Handling Enhancements • Incorporation of PeriodicDID Scheduler Type2 • Renaming of the SRS_Diagnostics to RS_Diagnostics • minor corrections / clarifications /editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published
Specification of Diagnostic Event Manager	• Diagnostic SRS renamed from SRS_DIAG_xxxx to RS_DIAG_xxxx • DataType of size parameter increased for FF and ED APIs • Inconsistencies and contradictions within the AR4.3.1 specifications of Dem and Dcm has been removed ▽

▽

△

Name	Specification history entry
	△ • Minor corrections / clarifications / editorial changes , For details please refer to the Change Documentation • Changed Document Status from Final to published
Specification of Diagnostic Log and Trace	• No Content Changes • Changed Document Status from Final to published
Specification of Diagnostic over IP	• Introduced CONC 649 DoIP Extension in draft state • Updated the functionality of routing activation for security use-cases • Increased multiplicity of DoIP target address so more than 255 DoIP addresses could be used • Minor corrections / clarifications / editorial changes; for details please refer to the ChangeDocumentation • Changed Document Status from Final to published
Specification of DIO Driver	• MCALMulticoreDistribution (CONC_639) • Changed Document Status from Final to published
Specification of ECU Configuration	• Updated specification to avoid usage of term MUST. • Specification of the format of the destinationType of an EcucForeignReferenceDef • Added support for Bsw Multicore Distribution in Ecuc module • Changed Document Status from Final to published
Specification of ECU Resource Template	• Editorial changes • Changed Document Status from Final to published
Specification of ECU State Manager	• No content changes • Changed Document Status from Final to published
Specification of EEPROM Abstraction	• Configuration layouts added • Changed Document Status from Final to published
Specification of EEPROM Driver	• MCAL Multicore Disctribution concept is changed from draft to Final • Changed Document Status from Final to published
Specification of Ethernet Driver	• 2500Mbit Ethernet Support • Eth_TimeStampQualType base type defined • Changed Document Status from Final to published
Specification of Ethernet Interface	• Some empty pages removed • API Table for EthIf_MainFunctionRx_<PriorityProcessing ShortName> corrected • EthSwt_PortModeType introduced to explicitly distinguish between Port Mode and Transceiver Mode • Missing and duplicate service IDs corrected ▽

▽

△

Name	Specification history entry
	△ • Missing API of EthSwt and EthTrcv are added in EthIf • "BSWDistribution" (CONC_643) added as draft • Changed Document Status from Final to published
Specification of Ethernet State Manager	• No Content Changes • Changed Document Status from Final to published
Specification of Ethernet Switch Driver	• Possibility to explicitly request or release Ethernet link state added • Replace usage of EthTrcv_ModeType with the Eth_ModeType • Support for 2500 MBit/s Ethernet connection • Fix Ethernet Hardware Initialization • Changed Document Status from Final to published
Specification of Ethernet Transceiver Driver	• 2500Mbit Ethernet Support • Cable Diagnostic • Uniformatisation Eth_Mode Type • Changed Document Status from Final to published
Specification of Extended Fixed Point Routines	• No content changes • Changed Document Status from Final to published
Specification of Fixed Point Interpolation Routines	• Editorial changes • Changed Document Status from Final to published
Specification of Fixed Point Math Routines	• Editorial changes • Changed Document Status from Final to published
Specification of Flash Driver	• Draft status of ECUC_Fls_00323 removed • Changed Document Status from Final to published
Specification of Flash EEPROM Emulation	• Added diagrams in chapter 10 • Added limitation about parallel access to Flash Driver • Changed Document Status from Final to published
Specification of Flash Test	• "DRAFT" status of ECUC_FlsTst_00175 removed • Changed Document Status from Final to published
Specification of FlexRay AUTOSAR Transport Layer	• No Content Changes • Changed Document Status from Final to published
Specification of FlexRay Driver	• Enhanced multi core usage support • Changed Document Status from Final to published
Specification of FlexRay Interface	• Clarification on handling of dynamic length LSdus • Changed Document Status from Final to published
Specification of FlexRay ISO Transport Layer	• Header file name changes in Chapter 8. • Changed Document Status from Final to published.
Specification of FlexRay Network Management	• No Content Changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of FlexRay State Manager	• Updated Transitions T03 and T06 • Changed Document Status from Draft to published
Specification of FlexRay Transceiver Driver	• Incorporation of validation results for [CONC_639] • Fix inconsistent renaming of general types headers • Bus-independent solution regarding channel states upon initialization introduced • Periodic Error Detection in Bus Driver added • Changed Document Status from Final to published
Specification of Floating Point Interpolation Routines	• Editorial changes • Changed Document Status from Final to published
Specification of Floating Point Math Routines	• Editorial changes • Changed Document Status from Final to • published
Specification of Function Inhibition Manager	• No content changes • Changed Document Status from Final to published
Specification of GPT Driver	• No content changes • Changed Document Status from Final to published
Specification of Hardware Test Manager on start up and shutdown	• Chapter 8 generated from BSW UML model • Changed Document Status from Final to published
Specification of I/O Hardware Abstraction	• EcuAbstractionComponentType changed to EcuAbstractionSwComponentType. • Changed Document Status from Final to published
Specification of ICU Driver	• Incorporation of validation results • Changed Document Status from Final to published
Specification of I-PDU Multiplexer	• Introduced BSW Distribution Concept (CONC_643) as draft • Changed Document Status from Final to published
Specification of Key Manager	• Editorial changes. • Create general error detection in chapter 7.4. • Changed Document Status from Final to published.
Specification of LIN Driver	• MCALMulticoreDistribution (CONC_639) • Changed Document Status from Final to published
Specification of LIN Interface	• No Content Changes • Changed Document Status from Final to published
Specification of LIN State Manager	• Editorial changes • Changed Document Status from Final to published
Specification of LIN Transceiver Driver	• MCALMulticoreDistribution (CONC_639) • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of MCU Driver	• Removed DRAFT status of items introduced for Multicore support • Removed duplicated chapters McuGeneralConfiguration and McuClockSettingConfig • Changed Document Status from Final to published
Specification of Memory Abstraction Interface	• Configuration layout added • Changed Document Status from Final to published
Specification of Memory Mapping	• Clarify NO-INIT policy • Clarify caseness of VendorApiInfix • Clarify usage of core scope • Update of referenced pictures • Changed Document Status from Final to published
Specification of Module E2E Transformer	• Incorporated usage of E2E_PxxForward methods to replicate detected E2E-Errors on outgoing messages • Added Client-Server Communication support • Updated Tracing from SRS_E2E to RS_E2E • Changed Document Status from Final to published
Specification of Module XCP	• No Content Changes • Changed Document Status from Final to published
Specification of Network Management for SAE J1939	• Clarification of AC transmission • Constraint regarding detection of address conflicts • Changed Document Status from Final to published
Specification of Network Management Interface	• Minor changes • Multicore Distribution support (draft) added • Changed Document Status from Final to published
Specification of NVRAM Manager	• Changes related to NVM_E_WRITE_PROTECTED • Port Prototypes are generated for block only if needed • Changed Document Status from Final to published
Specification of OCU Driver	• Error OCU_E_BUSY classifed as Runtime Error. • Added reference to OcuHWSpecificSettings in OcuChannel. Multiplicity of OcuHWSpecificSettings changed. • Introduced MCAL Multicore Distribution • Changed Document Status from Final to published
Specification of Operating System	• Various updates for ARTI • Enhanced memory mapping for IOC • Some type improvements for multi-core • Minor correction / clarification / editorial changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of PDU Router	• Add Multicore Distribution • Change SWS_PduR_00783 to process overlength PDUs • Add additional parameters in the PduRBswModules container • Changed Document Status from Final to published
Specification of Platform Types	• Editorial changes. • Wrong "Available via" references fixed. • Changed Document Status from Final to published.
Specification of Port Driver	• MCAL Multicore Distribution • Changed Document Status from Final to published
Specification of PWM Driver	• Introduced MCAL Multicore Distribution • Changed Document Status from Final to published
Specification of RAM Test	• MCALMulticoreDistribution (CONC_639) • Production errors updated • Editorial changes • Changed Document Status from Final to published
Specification of Secure Onboard Communication	• Added option to send default authentication information • Added an authentic PDU length header • Added new options to override the verification status • Minor corrections / clarifications / editorial changes; For details please refer to the Change Documentation • Changed Document Status from Final to published
Specification of Service Discovery	• Service activation depending on PNCs • Retry mechanism in combination with Cyclic Offers • EventGroup subscription updates from different servers • Clarification of SubscribeEventgroupNack handling • Changed Document Status from Final to published
Specification of Socket Adaptor	• Support for selectable PDU reception paths for multiple instances of the same Sd service • Changed Document Status from Final to published
Specification of SOME/IP Transformer	• Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding set to valid • Removed *_ACK message types • replaced implementsSOMEIPStringHandling (in class SOMEIPTransformationSignalProps) with implementsLegacyStringSerialization • Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of SPI Handler/Driver	• SWS_Spi_00082 removed • Changed Document Status from Final to published
Specification of Standard Types	• Added chapter Std_TransformerError • Editorial changes • Changed Document Status from Final to published
Specification of SW-C End-to-End Communication Protection Library	• Incorporated E2E_PxxForward methods to replicate detected E2E-Errors on outgoing messages • E2E_P0xSTATUS_ERROR values are now the same for all profiles • Fixed minor inconsistencies and typos • Updated Tracing from SRS_E2E to RS_E2E • Changed Document Status from Final to published
Specification of Synchronized Time-Base Manager	• Time Validation (draft) • Multicore Distribution support (draft) • Clarification regarding behavior when Time Stamp or User Data is invalid • Clarification on StbM 'Time Notifications' Feature • Changed Document Status from Final to published
Specification of TCP/IP Stack	• Introduction of IPsec • Minor corrections and clarifications • Changed Document Status from Final to published
Specification of Time Service	• No content changes • Changed Document Status from Final to published
Specification of Time Synchronization over CAN	• Time Validation (draft) • Clarification regarding messages with stuck sequence counter • Clarification regarding cyclic operation entry after timebase startup • Clarification regarding transmission and reception of User Bytes • Changed Document Status from Final to published
Specification of Time Synchronization over Ethernet	• Time Validation (draft) • Clarification regarding cyclic operation entry after timebase startup • Clarification regarding transmission and reception of User Bytes • Clarified SGW value handling for missing Sub-TLVs • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of Time Synchronization over FlexRay	• Time Validation (draft) • Clarification regarding messages with stuck sequence counter • Clarification regarding cyclic operation entry after timebase startup • Clarification regarding transmission and reception of User Bytes • Changed Document Status from Final to published
Specification of Timing Extensions	• Replaced the verb "must" by the verb "shall" according to TPS_STDT_00053. • Corrected spelling errors, terminology, and removed empty pages • Changed attributes of TDEventFrameEthernet to be consistent with changes applied to ethernet communications. • Corrected splitkey definition in timingResource. • Revised section "Occurrence Expression Language for Timing Events" to improve understanding of the purpose of complex timing description event, as well as the timing functions. • Changed Document Status from Final to published
Specification of TTCAN Driver	• Editorial changes • Changed Document Status from Final to published
Specification of TTCAN Interface	• Editorial changes • Changed Document Status from Final to published
Specification of UDP Network Management	• Det error handling corrected • Harmonization of API • Minor corrections • Changed Document Status from Final to published
Specification of Vehicle-2-X Basic Transport	• Added support for C2CCC BSP 1.3 • Changed Document Status from Final to published
Specification of Vehicle-2-X Facilities	• Added support for C2CCC BSP 1.3 • Editorial Changes • Service API bugs corrections • Changed Document Status from Final to published
Specification of Vehicle-2-X Geo Networking	• Added support for C2CCC BSP 1.3 • Changed Document Status from Final to published
Specification of Vehicle-2-X Management	• Update referenced Documents • Editorial changes • Changed Document Status from Final to published
Specification of Watchdog Driver	• Reworked items requiring that triggering of the WDG shall be done from interrupt routine • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Specification of Watchdog Interface	• Corrected error codes • Removed the error code WDGIF_E_INV_POINTER • Corrected uptrace of [SWS_WdgIf_00046] • Cleanup diagrams in chapter 10 • Changed Document Status from Final to published
Specification of Watchdog Manager	• Enhancement of Deadline Supervision to support timeout detection • Correction/Clarification of Supervision Algorithms and their configurations • Clarification of startup behaviour (incl. failed WdgIf_SetMode during init) • Corrected/Changed/Added Error Codes and other editorial issues • Changed Document Status from Final to published
Specification of Wireless Ethernet Driver	• Operation for DCC_Access queue modified • Partition handling released • Change Document Status from Final to published
Specification of Wireless Ethernet Transceiver Driver	• Partition handling released • Change Document Status from Final to published
Specification on SOME/IP Transport Protocol	• Editorial changes • Changed Document Status from Final to published
Specifications of Safety Extensions	• Deleted empty page • Changed Document Status from Final to published
Standardization Template	• harmonize the use of BlueprintCondition, FormalBlueprintGenerator • editorial changes • changed Document Status from Final to published
Supplementary material of general blueprints for AUTOSAR	• Update Multi dimensional ValueBlock • Changed Document Status from Final to published
SW-C and System Modeling Guide	• Editorial changes • Changed Document Status from Final to published
System Template	• Rework of Ethernet communication model • Added support for Signal-To-Service Translation • Added support for IPsec configuration • Minor corrections / clarifications /editorial changes; For details please refer to the ChangeDocumentation • Changed Document Status from Final to published
Unique Names for Documentation, Measurement and Calibration: Modeling and Naming Aspects including Automatic Generation	• No content changes • Changed Document Status from Final to published
Utilization of Crypto Services	• No content changes • Changed Document Status from Final to published

▽

△

Name	Specification history entry
Virtual Functional Bus	• No content changes • Changed Document Status from Final to published
XML Schema Production Rules	• Editorial changes • Changed Document Status from Final to published
XML Specification of Application Interfaces	• Add and update TRSM Domain content regarding Hybrid drivetrain systems • Changed Document Status from Final to published

Table 5.1: Release History