Document Title	Adaptive Platform Release Overview
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	782

Document Status	Final
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	18-10
Release Life Cycle Status	Initial Release

Document Change History			
Date	Release	Changed by	Change Description
2018-10-31	18-10	AUTOSAR Release Management	Initial release

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview

Table of Contents

1 Introduction

1.1 Scope of this document

This document provides an overview on the AUTOSAR standard Adaptive Platform Release 18-10.

1.2 Dependencies to other standards

This release of the Adaptive Platform depends on the standard Foundation in Release 1.5.0, which

- defines protocols implemented by Adaptive Platform and
- contains the project objectives and the common requirements from which the features of the Adaptive Platform are derived.

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

1.3 Content of chapters

This document is structured as follows:

- Chapter 2 provides an introduction to the Adaptive Platform and its standardization approach.
- Chapter 3 provides a list of documentation references.
- Chapter 4 contains the overview of specifications comprising the Release 18-10. This chapter is structured according to the clusters of AUTOSAR Release 18-10.
- Chapter 5 provides a summary of changes since the previous release of the Adaptive Platform.
- Chapter 6 contains remarks about known technical deficiencies.
- Chapter 7 contains the detailed release history of all released specifications.
- Chapter 8 provides a set of definitions aimed to increase the understanding of the content of this document and the Release 18-10.

2 Introduction to the Adaptive Platform

The AUTOSAR Adaptive Platform is the standardized platform for microprocessor-based ECUs supporting use cases like highly automated driving as well as high speed on-board and off-board communication.

The Adaptive Platform differs in a number of aspects from the standardization approach of the Classic Platform:

- Release strategy
- Parallel validation of specification via software implementation
- Specification of functional clusters instead of modules

2.1 Release strategy

The Adaptive Platform will be in lifecycle state "Development" according to AUTOSAR's lifecycle model for its standards (see figures 1 and 2) until it will reach a certain maturity level. Until October 2019, AUTOSAR will release a new version of the Adaptive Platform in a 6-month cycle. The lifecycle state "Development" implies that users of the Adaptive Platform have no guarantee on backward compatibility. Consequently, all requirements have the lifecycle status *draft*. For items in specifications this attribute is not explicitly set but is the default state.

This release is a joint release with the Classic Platform R4.4.0 and thus the major focus of this release lies in the synchronization with the Classic Platform.
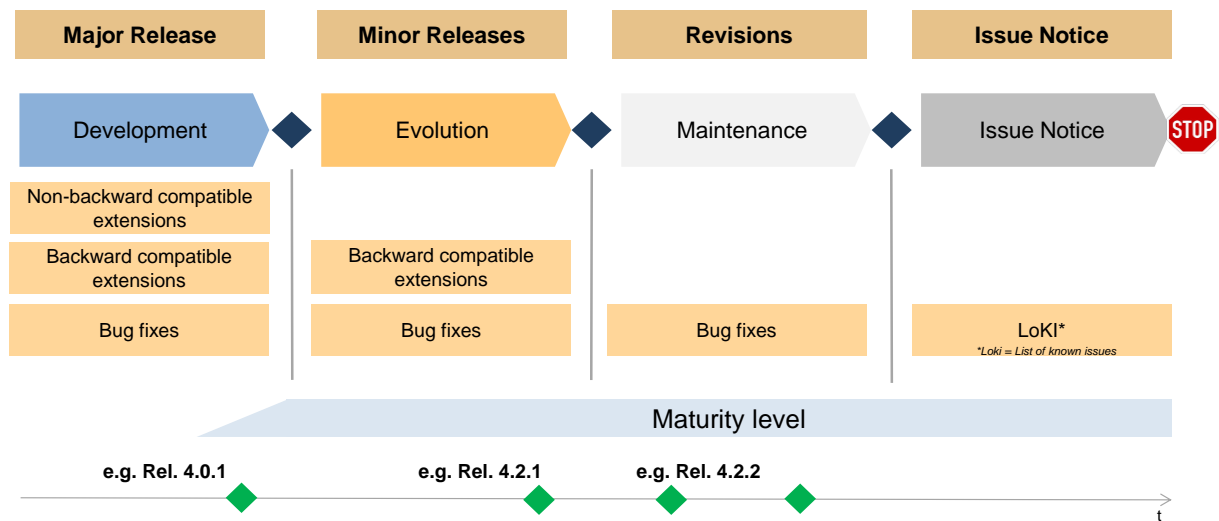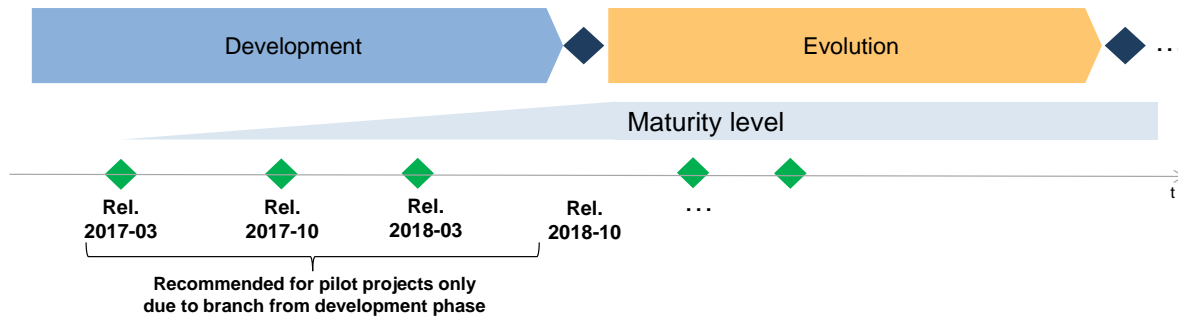

Figure 1: Lifecycle model of AUTOSAR standards and its application to Classic Platform

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview

**Figure 2: Application of AUTOSAR lifecycle to Adaptive Platform
(Release numbers only exemplary)**

Apart from the regular specifications that have been elaborated in intensive expert discussion, according to current planning, the releases may comprise draft specifications to indicate the intended scope and direction of discussion to the AUTOSAR development community.

The following must be considered for the draft specifications:

1. Minimal or less quality measures are being applied during development
2. No indication of quality / stability due to a lack of discussions between the AUTOSAR partners

As new features will be added with each release, AUTOSAR encourages the partners to contribute in the development of the features.

2.2 Parallel validation of specification via implementation

The Adaptive Platform is validated through an AUTOSAR-internal implementation: the Adaptive Platform Demonstrator. This Demonstrator is available to all the partners and can be a reference to understand the underlying concepts of the Adaptive Platform. The Demonstrator provides an implementation example based on the specification rather than a reference implementation. All further development based on the Demonstrator will become the responsibility of the respective partner. For legal constraints see the dedicated paragraphs in the Development Agreement.

For the current releases, the Demonstrator software implementation has undergone only informal reviews with no strict quality assurance. AUTOSAR is increasing the quality assurance significantly to ensure the maintainability and extensibility of the Demonstrator software implementation.
The Demonstrator comes with traceability back to the specifications and explanatory documents or the so-called Functional Cluster Design specifications.
Additionally AUTOSAR develops System Tests to test the demonstrator implementation against the AUTOSAR requirements. These tests are also part of the release.

The Adaptive Platform source code for 18-10 is meant to be a validation of the released specifications and will therefore be released shortly after the specifications.

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview
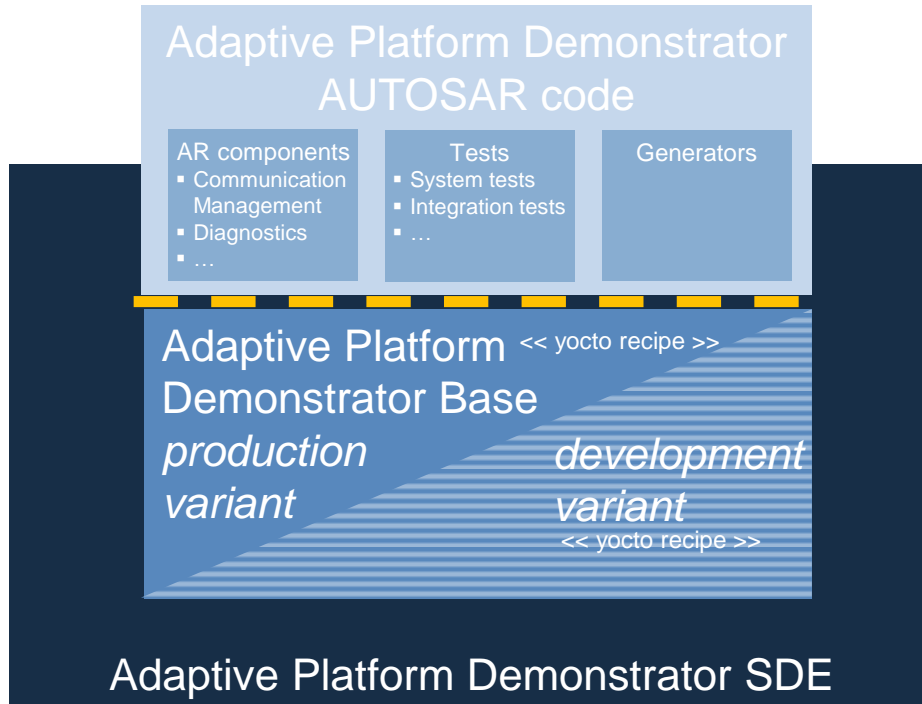
Figure 3: Overview of the AUTOSAR Adaptive Platform Demonstrator

2.3 Specification depth

Based on the development history of the Classic Platform, AUTOSAR has decided to specify functional clusters instead of a specific software architecture to provide the implementers with options to find efficient solutions for the standardized features.

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview

3 Related documentation

[1] AUTOSAR Specifications in general
[2] Glossary

4 Specification overview

The published specifications are divided into the following clusters:

- Release Documentation
- General
- Methodology and Manifests
- Adaptive Foundation
- Adaptive Services

The assignment of specifications to clusters is shown below.

Long Name	File Name	Life cycle changes	Draft Specification
Release Documentation			
Adaptive Platform Release Overview	AUTOSAR_TR_Adaptive PlatformReleaseOverview		
AUTOSAR Adaptive Platform Specification Hashes	AUTOSAR_TR_Adaptive PlatformSpecificationHashes		
Adaptive Foundation			
Explanation of ara::com API	AUTOSAR_EXP_ARACom API		
Explanation of IPsec Implementation Guidelines	AUTOSAR_EXP_IPsecImpl ementationGuidelines	Initial release	
Requirement on Time Synchronization for Adaptive Platform	AUTOSAR_RS_TimeSynch ronization		
Requirements on Communication Management	AUTOSAR_RS_Communic ationManagement		
Requirements on Cryptography	AUTOSAR_RS_Cryptograp hy		
Requirements on Execution Management	AUTOSAR_RS_Execution Management		
Requirements on Identity and Access Management	AUTOSAR_RS_IdentityAnd AccessManagement		
Requirements on Operating System Interface	AUTOSAR_RS_OperatingS ystemInterface		
Requirements on Persistency	AUTOSAR_RS_Persistenc y		
Requirements on Platform Health Management for Adaptive Platform	AUTOSAR_RS_PlatformHe althManagement		
Requirements on Security Management for Adaptive Platform	AUTOSAR_RS_SecurityMa nagement		
Specification of Communication Management	AUTOSAR_SWS_Commun icationManagement		

Long Name	File Name	Life cycle changes	Draft Specification
Specification of Cryptography for Adaptive Platform	AUTOSAR_SWS_Cryptography		X
Specification of Execution Management	AUTOSAR_SWS_ExecutionManagement		
Specification of Identity and Access Management	AUTOSAR_SWS_IdentityAndAccessManagement		
Specification of Log and Trace	AUTOSAR_SWS_LogAndTrace		
Specification of Operating System Interface	AUTOSAR_SWS_OperatingSystemInterface		
Specification of Persistency	AUTOSAR_SWS_Persistency		
Specification of Platform Health Management for Adaptive Platform	AUTOSAR_SWS_PlatformHealthManagement		
Specification of RESTful communication	AUTOSAR_SWS_REST		
Specification of Time Synchronization for Adaptive Platform	AUTOSAR_SWS_TimeSynchronization		
Adaptive Services			
Explanation of Sensor Interfaces	AUTOSAR_EXP_SensorInterfaces	Initial release	
Requirements of State Management	AUTOSAR_RS_StateManagement	Initial release	
Requirements on Update and Configuration Management	AUTOSAR_RS_UpdateAndConfigManagement		
Specification for Network Management	AUTOSAR_SWS_NetworkManagement		
Specification of Diagnostics	AUTOSAR_SWS_Diagnostics		
Specification of State Management	AUTOSAR_SWS_StateManagement	Initial release	
Specification of Update and Configuration Management	AUTOSAR_SWS_UpdateAndConfigManagement		
Methodology and Manifests			
Meta Model	AUTOSAR_MMOD_MetaModel		
Meta Model-generated XML Schema	AUTOSAR_MMOD_XMLSchema		
Methodology for Adaptive Platform	AUTOSAR_TR_AdaptiveMethodology		
Requirements on Manifest Specification	AUTOSAR_RS_ManifestSpecification		

Long Name	File Name	Life cycle changes	Draft Specification
Specification of Manifest	AUTOSAR_TPS_ManifestSpecification		
Specification of Platform Types for Adaptive Platform	AUTOSAR_SWS_AdaptivePlatformTypes		
Supplementary material of the AUTOSAR XML Schema	AUTOSAR_TR_XMLSchemaSupplement		
Collection of blueprints for AUTOSAR Adaptive Platform models	AUTOSAR_MOD_AdaptivePlatformGeneralBlueprints	Initial release	
General			
Design guidelines for using parallel processing technologies on Adaptive Platform	AUTOSAR_EXP_ParallelProcessingGuidelines		
Explanation of Adaptive Platform Design	AUTOSAR_EXP_PlatformDesign		
Explanation of Safety Overview	AUTOSAR_EXP_SafetyOverview		
Functional Cluster Shortnames	AUTOSAR_TR_FunctionalClusterShortnames		
General Requirements specific to Adaptive Platform	AUTOSAR_RS_General		
General Specification of Adaptive Platform	AUTOSAR_SWS_General		
Guidelines for the use of the C++14 language in critical and safety-related systems	AUTOSAR_RS_CPP14Guidelines		
Guidelines for using Adaptive Platform interfaces	AUTOSAR_EXP_AdaptivePlatformInterfacesGuidelines	Initial release	
Specification of Core Types for Adaptive Platform	AUTOSAR_SWS_CoreTypes		
System Tests of Adaptive Platform	AUTOSAR_TR_AdaptivePlatformSystemTests		

All specifications contain requirements which are identified by special braces:
⌈ = Beginning of the requirement content
⌋ = End of the requirement content
In addition, all XML files and schemas are considered as requirements.

5 Summary of changes

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

- Regular maintenance of document.
- New Documents have been created and are released for the first time.
- Documents went through a major rework.

5.1 Release 18-10

In AUTOSAR R18-10 the Adaptive Platform has been elaborated further to harmonize with the Classic Platform.

Additionally the System Tests have been reworked to test the Adaptive Platform Demonstrator against the Requirements Specifications of the AUTOSAR Adaptive Platform.

Further improvements have been applied to all the specifications that are part of R18-10.

5.1.1 Concepts

The following concepts in 5.1.1.1 – 5.1.1.2 have been introduced.

5.1.1.1 Formal Model Query and Blueprint Derivation Mechanisms

The concept "Formal Model Query and Blueprint Derivation Mechanisms" is released as draft and will be validated in 2019.

The concept completes the extension of AUTOSAR Classic (CP) and Adaptive platforms (AP) with the AUTOSAR Model Query Language (ARMQL). This new language enables a highly efficient collaboration of AUTOSAR user due to resolving variation points in CP and AP by the same mechanism. It is published in textual form, not bound to a specific tool and significant better understandable as the existing Formula Language.

5.1.1.2 Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding (TLV)

The concept TLV is released as draft and will be validated in 2019.

The concept adds support for improved forward and backward compatibility during evolution of interfaces on SOME/IP protocol-level. Moreover, the concept integrates support for optional struct members on protocol-level and application-level (RTE and ara::com).

5.1.2 Specifications

5.1.2.1 New Specifications

The following documents and templates were added to the R18-10:

- Specification of State Management (UID 908, SWS)
- Requirements of State Management (UID 909, RS)
- Explanation of Sensor Interfaces (UID 913, EXP)
- Guidelines for using Adaptive Platform interfaces (UID 929, EXP)
- Explanation of IPsec Implementation Guidelines (UID 930, EXP)
- Collection of blueprints for AUTOSAR Adaptive Platform models (UID 931, MOD)

5.1.2.2 Migrated Specifications

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

- Requirements on Adaptive Network Management (UID 898, RS), merged with the new document Requirements on Network Management (UID 927, RS)

5.1.2.3 Obsolete Specifications

The following specification is set to status "obsolete" in this release:

- No specifications were set to "obsolete".

5.1.2.4 Cancelled Specifications

The following specification is canceled in this release:

- No specifications were canceled.

5.1.3 Release Documentation

There were no major changes regarding the Release Documentation.

6 Remarks to known technical deficiencies

The technical deficiencies per specification are – if applicable – mentioned inside the respective specification in a chapter called "Known Limitations".

There are the following technical deficiencies which are not related to a particular specification: None

6.1 Known technical deficiencies per document

Document Long Name	Known Limitations
Methodology for Adaptive Platform	The following sections are still under discussion: • Section 2.4.5 (Set up an initial Machine), • Section 2.4.6 (Create Software Packages) • Section 2.4.7 (Management and provision of Software Packages)

Specification of Communication Management	• Local Buffer Overruns: Currently it is not specified what happens if local buffers are full because the application accesses data slower than they are received over the network.
	• The Signal to Service mapping in this specification does not contain behavior specification.
	• The E2E communication protection works only for events which are polled and which are transmitted at least once per fault tolerant time interval. This means, it requires:
	  o Periodic invocation of the method Update in a polling mode
	  o Periodic or mixed-periodic invocation of the method Send
	  o In case Update or Send are not invoked periodically, then some communication failure modes are not detected (loss, delay and possibly also repetition). In this case, if E2E is used, then additional measures need to be taken at application level to address those non-detected failure modes.
	• The values of some E2E parameters are defined by the standard and shall not be changed.
	• Optional method arguments: The Specification does not support the existence of optional method arguments
	• Some limitations apply for optional arguments introduced with the TLV serialization

Document Long Name	Known Limitations
Specification of Core Types for Adaptive Platform	• The specification of some data types (Array, Map, Optional, String, StringView, Variant) mentions "supporting constructs", but lacks a precise scope definition of this term. • The specification of some data types (Map, Vector, String) is lacking a comprehensive definition of memory allocation behavior; it currently only describes it as "implementation-defined". • Chapter 7 ("Functional Specification") describes some behavior informally that should rather be given as specification items.
Specification of Cryptography for Adaptive Platform	• The entire Crypto API is provisional and likely to be fully revised in the upcoming releases. • The content was not updated for the current release.
Specification of Diagnostics	• It is probable that shared_ptr will change in an upcoming release.
Specification of Execution Management	• Support for Resource Limitation is not complete • Support for Fault Tolerance (reacting to, and coping with errors in EM itself) is not complete • Support for establishment of a Trusted Platform ([RS_EM_00014]) is not specified
Specification of Identity and Access Management	• The topic of providing identity information of Adaptive Applications to PEPs is still under discussion. • Requirements and specification details regarding Application ID / Application Instance ID and providing application identity in general may be affected by this discussion and may change accordingly. • There is no API specification available yet
Specification of Manifest	• The AUTOSAR SWS REST [5] defines a low-level API for REST-based communication. • The content of section 11, on the other hand, applies for the configuration of a not-yet standardized API on top of the ara::rest API.
Specification of Network Management	• Only supports UdpNM at the moment • Does not allow node detection - Repeat Message State requests but handles incoming requests • Cannot be configured as master network coordinator • Does not support coordinated shutdown using information from CBV • Does not support passive mode or passive startup • Mapping of logical networks to BitVector positions (in the message) not available in the manifest • New: o User data cannot be accessed from applications in a standardized way – the service interface was moved as NM now interacts with SM instead of applications. User data has to be clarified for next release.

Document Long Name	Known Limitations
Specification of Operating System Interface	• There is currently no (sufficient) API providing periodic time-based processing to fulfill [RS_OSI_00102]. • Authorized access to APIs is not supported [RS_OSI_00205, RS_OSI_00208].
Specification of Persistency	• The interpretation of deployment related information in the AUTOSAR model is not yet covered in detail in this specification. In addition, the concept of a roll-back after an update is not yet supported. • The configuration of encryption for Persistency is not defined in [2].
Specification of Platform Health Management for Adaptive Platform	• Daisy chaining (i.e. forwarding Supervision Status, Checkpoint or Health channel information to an entity external to PHM or another PHM instance) is currently not supported in this document release. • Platform Health Management configuration related to Supervision Modes is not fully supported in this document release. • An API to inform Supervised Entities about the Supervision states is available only in polling mode. No API using notification mode is available in this release. • Interface with the Diagnostic Manager is not specified in this release.
Specification of RESTful communication	• The interfaces are only specified to the point to make semantics clear • The error handling for RESTful communication is currently limited due to the fact that errors are not reported in the context of a request transmission
Specification of State Management	• Section 7.3 on Component States are partially discussed and not finished yet. • The RequestRestart and Communication Control for Diagnostic reasons is a proposal only and subject to change.
Specification of Time Synchronization for Adaptive Platform	• The Time Synchronization module is bound to Adaptive Platform Systems. • For the TS, it is necessary that at least there is one TBR in the system, otherwise no functionality can be provided to the Adaptive Applications • API design is not fully compliant to Adaptive Platform Design Rules which request the usage of UpperCamelCase.

Document Long Name	Known Limitations
System Tests of Adaptive Platform	- Test cases may not cover whole RS as specified against test cases - Test setup figure may not exactly reflect the test configuration - Test cases may not be fully covered by corresponding system test implementations - System test cases are just examples, since there could be many ways to define and implement use case scenarios - DIAG does not have any RS traceability, as it is intended to reuse WP-T results - LT does not have any RS traceability. Traceability will be added in next release

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview

7 Release history

7.1 Release 18-10

Release 18-10 was originally released on the 31st of October 2018.

Name	Specification history entry
Design guidelines for using parallel processing technologies on Adaptive Platform	• Minor changes
Explanation of Adaptive Platform Design	• Changes to reflect the latest SWS contents
Explanation of ara::com API	• Added InstanceIdentifier and InstanceSpecifier explanation • Restructured chapter structure • Adapt FindService signatures • Added sample code for event usage • Restructured chapter structure • Proxy and skeleton instances are not copyable • Changed certain data types to ara::core namespace. • Adapted to new error handling based on ara::core::ErrorCode
Explanation of Automated Driving Interfaces	• Initial release
Explanation of Ipsec Implementation Guidelines	• Initial release
Explanation of Safety Overview	• Restructuring of document inspired by ISO 26262 • Rework chapters 1-5 • Add functional safety requirements table
Functional Cluster Shortnames	• Renaming of Identity and Access Management, Operating System Interface, Update and Configuration Management
General Requirements specific to Adaptive Platform	• More details to clause 1 Scope of document given • Former chapter 4.3 on Design requirements putted below chapter 4.2 Non-functional requirements • Following requirements have been revised: [RS_AP_00111], [RS_AP_00113], [RS_AP_00114], [RS_AP_00115], [RS_AP_00122], [RS_AP_00120], [RS_AP_00121], [RS_AP_00124], [RS_AP_00125] • Following requirements have been deleted: [RS_AP_00117], [RS_AP_00118] • Following requirements have been added: [RS_AP_00127], [RS_AP_00128], [RS_AP_00129], [RS_AP_00130], [RS_AP_00131], [RS_AP_00132], [RS_AP_00134]
General Specification of Adaptive Platform	• SWS_AP_00003 removed since there is no demand in RS_AP_00003 which requires it anymore
Guidelines for the use of the C++14 language in critical and safety-related systems	• Added traceability for ISO 26262 (B.6) • New rules resulting from continued analysis of the C++ Core Guideline • Finished addressing MISRA review comments of the 2017-03 release • Improvements of already existing rules, more details in the Changelog (D.3) • Marked the specification as obsolete
Guidelines for using Adaptive Platform interfaces	• Initial release

Name	Specification history entry
Methodology for Adaptive Platform	• Renamed Application Manifest to Execution Manifest • Moved references from spec.item body to foot notes • Editorial changes
Requirement on Time Synchronization for Adaptive Platform	• Minor changes and bugfixes • Editorial changes
Requirements of State Management	• Initial release
Requirements on Communication Management	• Minor changes and bugfixes
Requirements on Cryptography	• Removed: [RS_CRYPTO_02303] and [RS_CRYPTO_02402] • Updated: [RS_CRYPTO_02006]
Requirements on Execution Management	• Removed: RS_EM_00003, RS_EM_00004, RS_EM_00110 and RS_EM_00111. • Added: [RS_EM_00014]
Requirements on Identity and Access Management	• Functional Description of Capabilities • Functional Description of Access • Control for Inter-Platform Communication • Requirement for Superset Manifests
Requirements on Manifest Specification	• Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation in AUTOSAR_RS_ManifestSpecification
Requirements on Operating System Interface	• Removed: RS_OSI_00102 and RS_OSI_00105 • Added: [RS_OSI_00207], [RS_OSI_00208].
Requirements on Persistency	• Restructured document • Added resource handling • Added UCM related requirements
Requirements on Platform Health Management for Adaptive Platform	• minor corrections / clarifications / editorial changes
Requirements on Security Management for Adaptive Platform	• Chapter 2.3 'Protected Runtime Environment' revised
Requirements on Update and Configuration Management	• Requirements on Operating System updates • Requirement on Security • Requirement on History
Specification of Communication Management	• Introduced Adaptive Core types • Introduced exception-less API • Refined DDS network binding • Minor changes and bugfixes
Specification of Core Types for Adaptive Platform	• Add chapter 2 with acronyms • Add chapter 4 with limitations of the current specifications • Add chapter 5 with dependencies to other modules • Add chapter 7 • Add classes representing the approach to error handling to chapter 8 • Adapt classes Future and Promise to the error handling approach • Add global functions for initialization and shutdown of the framework • Add class InstanceSpecifier to chapter 8 • Add more types and functions from the C++ standard

Name	Specification history entry
Specification of Cryptography for Adaptive Platform	• No changes
Specification of Diagnostics	• Diagnostic Protocol replaced by Diagnostic Conversations • ResponseOnEvent, CommunicationControl, EcuReset added • Chapter 7 overall rework and updates • Chapter 8 split into chapter 8 (C++ API) and chapter 9 (Service Interfaces)
Specification of Execution Management	• Refinement of Deterministic Execution • Updated Process lifecycle to clarify • Process and Execution States • Updated Application Recovery Actions
Specification of Identity and Access Management	• Reworked functional specification • Removed API specification for general rework
Specification of Log and Trace	• Changed initialization APIs • Improved references • Log file definition
Specification of Manifest	• Finish introduction of CppImplementationDataType • Support for optional elements in structures • Rework configuration of adaptive platform modules
Specification of Network Management	• Updated interaction with State Management • Removed APIs and Services (interaction is done via SM) • Temporary removed user data access to applications
Specification of Operating System Interface	• Add Resource Control • Added Shared object support
Specification of Persistency	• Introduction of ara::core types and switch to exceptionless API • Rework of redundancy approach • Support for resource limitation • Improvements and harmonization of KeyValueStorage and FileProxy API
Specification of Platform Health Management for Adaptive Platform	• Described the interfaces with functional clusters execution management and state management
Specification of Platform Types for Adaptive Platform	• Rework to CppImplementationDataTypes
Specification of RESTful communication	• Updated APIs to use ara::core types • Minor editorial fixes
Specification of State Management	• Initial release
Specification of Time Synchronization for Adaptive Platform	• Minor changes and bugfixes • Editorial changes
Specification of Update and Configuration Management	• Updated interaction other functional clusters like PER and EMO/SM • Introduction of vehicle package distribution
System Tests of Adaptive Platform	• Added RS traceability for test cases • Added ISO 9646 framework and mapping on system test architecture • Added more test cases for CM, REST, EMO, and UCM

Document ID 782: AUTOSAR_TR_AdaptivePlatformReleaseOverview

8 Appendix

8.1 Definitions

See [3] for AUTOSAR definitions.

8.1.1 Release number

AUTOSAR applies a four-digit numbering scheme Ryy-mm to identify releases.
- yy   =   year
- mm   =   month

8.1.2 Specification item and requirement life cycle states

Specification items: The life cycle state is 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.
- **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. This is the default for Adaptive Platform.
- **Obsolete:** This indicates that the related entity is obsolete and will be removed in the next release.

If there is no life cycle state information stated then the state is Draft.

Requirements: The requirement attribute "type" indicates the life cycle state of the requirement. The states are the same as the specification item states.