Document Title	General Specification of Adaptive Platform
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	715

Document Status	Final
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	19-03

Document Change History			
Date	Release	Changed by	Description
2019-03-29	19-03	AUTOSAR Release Management	• No content changes.
2018-10-31	18-10	AUTOSAR Release Management	• SWS_AP_00003 removed since there is no demand in RS_AP_00003 which requires it anymore.
2018-03-29	18-03	AUTOSAR Release Management	• Description and supporting material changed for SWS_AP_00001 and SWS_AP_00002
2017-10-27	17-10	AUTOSAR Release Management	• Minor fixes
2017-03-31	17-03	AUTOSAR Release Management	• Initial release

Disclaimer

This work (specification and/or software implementation) and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the work.

The material contained in this work is protected by copyright and other types of intellectual property rights. The commercial exploitation of the material contained in this work requires a license to such intellectual property rights.

This work may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only. For any other purpose, no part of the work may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The work has been developed for automotive applications only. It has neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Table of Contents

1 Scope of this document

The goal of this document is to define a common set of basic requirements that apply to all functional clusters of the Adaptive Platform. It adds a common part to the specifications and it needs to be respected by platform vendors.

1.1 Document Conventions

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template, chapter Support for Traceability ([1]).

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see Standardization Template, chapter Support for Traceability ([1]).

2 Acronyms and Abbreviations

There are no acronyms and abbreviations relevant within this document that are not included in the [2, AUTOSAR glossary].

3 Requirements Tracing

The following tables reference the requirements specified in [3] and links to the fulfillment of these. Please note that if column "Satisfied by" is empty for a specific requirement this means that this requirement is not fulfilled by this document.

Requirement	Description	Satisfied by
[RS_AP_00116]	Header file name.	[SWS_AP_00001]

4 Requirements specification

4.1 Non-functional Requirements

[SWS_AP_00001]{DRAFT} **Include folder structure** ⌈

Type:	draft
Description:	All #include directives in header files that refer to ARA libraries shall be written in the form `#include "ara/fc/header.h"` with "ara" as the first path element, "fc" being the remaining directory path of the implementation's *installed* header file, starting with the Functional Cluster short name, and "header.h" being the filename of the header file.
Rationale:	Harmonized look and feel.
Dependencies:	–
Use Case:	–
Supporting Material:	The Functional Cluster short names are defined in AUTOSAR_TR_FunctionalClusterShortnames.pdf. Example: Execution Management (short name "exec") provides class `ApplicationClient`, which can be accessed with: `#include "ara/exec/application_client.h"` The "..." form of #include statements shall be used, due to the recommendation given in ISO/IEC 14882:2014 section 16.2.7.

⌋*(RS_AP_00116)*

[SWS_AP_00002]{DRAFT} **Prevent multiple inclusion of header file** ⌈

Type:	draft
Description:	All public header files shall prevent multiple inclusion by using #include guards that are likely to be system-wide unique.
Rationale:	Common coding guideline
Dependencies:	–
Use Case:	–
Supporting Material:	While uniqueness can generally not be guaranteed, the likelihood of collisions can be decreased with a naming scheme that is regular and results in long symbol names. The following #include guard naming scheme shall be used by implementations for all header files that cover symbols within the `ara` namespace or a sub-namespace therein: `ARA_<PATH>_H_` ▽

▽

△

<table>
<tr><td></td><td>

△

where <PATH> is the relative path name of the header file within the location of the implementation's *installed* header files, starting with the Functional Cluster name (and omitting the file extension), and with all components of <PATH> separated by underscore ("_") characters and containing only upper-case characters of the ASCII character set.

Example: The header file included with `#include "ara/log/logger.h"` shall use the #include guard symbol `ARA_LOG_LOGGER_H_`.

</td></tr>
</table>

⌋*()*

5 References

[1] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[2] Glossary
AUTOSAR_TR_Glossary

[3] General Requirements specific to Adaptive Platform
AUTOSAR_RS_General