Document Title	Requirements on Operating System Interface
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	718

Document Status	published
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	R20-11

Document Change History

Date	Release	Changed by	Description
2020-11-30	R20-11	AUTOSAR Release Management	• Uptrace to RS_Safety[1] document • Clarified Execution Management description • Moved time-triggered execution to RS_ExecutionManagement[2]
2019-11-28	R19-11	AUTOSAR Release Management	• Updated document structure • Changed Document Status from Final to published
2019-03-29	19-03	AUTOSAR Release Management	• Added: use case for [RS_OSI_00201] and [RS_OSI_00202]
2018-10-31	18-10	AUTOSAR Release Management	• Removed: RS_OSI_00102 and RS_OSI_00105 • Added: [RS_OSI_00207], [RS_OSI_00208].
2018-03-29	18-03	AUTOSAR Release Management	• Removed: RS_OSI_00101, RS_OSI_00200 and RS_OSI_00205. • Added: [RS_OSI_00103].
2017-10-27	17-10	AUTOSAR Release Management	• Minor changes, document clean up
2017-03-31	17-03	AUTOSAR Release Management	• Initial release

Disclaimer

Table of Contents

1 Scope of This Document

This document specifies the requirements of AUTOSAR Adaptive Platform on the Operating System that is part of the Foundation in the AUTOSAR Adaptive Platform.

2 Conventions to be Used

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template [3], chapter Support for Traceability.

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see Standardization Template [3], chapter Support for Traceability.

2.1 Requirements Guidelines

2.1.1 Requirements Quality

[RS_OSI_NA]{DRAFT} ⌈These requirements are not applicable as they are not within the scope of this release.⌋ *(RS_Main_00026, RS_Main_00030, RS_Main_00080, RS_Main_00140, RS_Main_00160, RS_Main_00161, RS_Main_00180, RS_Main_00190, RS_Main_00230, RS_Main_00250, RS_Main_00260, RS_Main_00261, RS_Main_00270, RS_Main_00280, RS_Main_00300, RS_Main_00301, RS_Main_00310, RS_Main_00320, RS_Main_00340, RS_Main_00350, RS_Main_00360, RS_Main_00440, RS_Main_00445, RS_Main_00480, RS_Main_00490, RS_Main_00491, RS_Main_00500, RS_Main_00501, RS_Main_00507, RS_Main_00510, RS_Main_00511, RS_Main_00650, RS_Main_00652, RS_Main_00653, RS_Main_01001, RS_Main_01002, RS_Main_01003, RS_Main_01004, RS_Main_01005, RS_Main_01007, RS_Main_01008, RS_Main_01025, RS_Main_01026)*

2.1.2 Requirements Identification

2.1.3 Requirements Status

The following requirements are described within this document but not otherwise considered in this release:

- [RS_OSI_00204]
- [RS_OSI_00208]

The functionality described above is subject to modification and will be considered for inclusion in a future release of this document.

3 Acronyms and Abbreviations

Abbreviation / Acronym:	Description:
Operating System Interface	A Functional Cluster within the Adaptive Platform Foundation.
AUTOSAR Adaptive Platform	see [4] AUTOSAR Glossary
Adaptive Platform Foundation	see [4] AUTOSAR Glossary
Adaptive Application	see [4] AUTOSAR Glossary
Execution Management	The element of the AUTOSAR Adaptive Platform responsible for the ordered startup and shutdown of the AUTOSAR Adaptive Platform and Adaptive Applications.
Application	see [4] AUTOSAR Glossary
Operating System	Software responsible for managing Processes on a Machine and for providing an interface to hardware resources.
Machine	see [4] AUTOSAR Glossary
Process	see [4] AUTOSAR Glossary
Functional Cluster	see [4] AUTOSAR Glossary

4 Requirements Specification

4.1 Functional Overview

The Operating System is responsible for run-time resource management (including time) for all Applications on and within the AUTOSAR Adaptive Platform. This includes not only the Adaptive Applications that run on top of ARA provided by AUTOSAR Adaptive Platform, but also the Functional Clusters that constitute the platform, which are also implemented as Applications. The OS functions in cooperation with Execution Management which is responsible for platform initialization and the start-up / shut-down of Applications.

Note that this Operating System Interface (OSI) requirement specification contains two different categories. The first category contains the requirements that are directly needed by the Adaptive Applications. The other category contains the ones that are needed by the AUTOSAR Adaptive Platform to realize implementation of Functional Clusters, especially the required mechanisms are difficult or inefficient to be implemented by other software entity than the OS. The most notable such Functional Cluster requiring the various OS mechanisms is Execution Management.

4.2 Functional Requirements

This chapter describes all requirements driving the work to define the Operating Systems functionality.

4.2.1 Assumption of Use

This section describes execution environment of the AUTOSAR Adaptive Platform that is directly or indirectly used to run the Applications. This execution environment is not requirements to the AUTOSAR Adaptive Platform in the strict sense, but rather assumptions on properties of Application running on the AUTOSAR Adaptive Platform. These assumptions are used to motivate the further requirements, and provide hints for Application developers to check if their use cases are covered in this specification document and which specific requirements are derived from those use cases.

The Operating System section defines requirements on the Operating System that processes can consider fulfilled in order to achieve their function.

4.2.2 General Requirements

This section describes APIs that should be exposed to processes on the AUTOSAR Adaptive Platform that closely relates to the Operating System functionality. Some libraries in the system may be considered by a Developer to be part of the Operating System without strictly-speaking belonging to it, for example the C++ runtime libraries. For the purpose of this section, the Operating System kernel and its libraries are considered to be a single entity.

[RS_OSI_00100] The Operating System Interface provided to processes shall provide a PSE51-compliant API. ⌈

Type:	valid
Description:	The foundation of the Operating System provided to the process shall be POSIX-compliant as defined by PSE51.
Rationale:	The defined functionality of the POSIX profile PSE51 defined by IEEE1003.13 [5] is provided by various off-the-shelf operating systems. The PSE51 profile is intended for embedded systems, with a single multi-threaded process, no file system, no user and group support and only selected options from more general IEEE1003.1 [6], which is the well-known POSIX standard. PSE51 offers functions for basic synchronized I/O, high-resolution timer, signals, ▽

▽

△

	△ semaphores, shared memory and threads. As the envisioned Application software components will not require to fork new processes themselves, and only need limited direct access to files, the PSE51 profile is thought to be sufficient.
Dependencies:	–
Use Case:	Application portability.
Supporting Material:	IEEE1003.13 [5] and IEEE1003.1 [6]

⌋(*RS_Main_00002*, *RS_Main_00050*, *RS_Main_00150*, *RS_Main_00420*)

[RS_OSI_00103] The Operating System Interface shall support C++. ⌈

Type:	valid
Description:	The Operating System interface shall support C++11.
Rationale:	Processes are written in C++ and interfaces are expected to conform to C++11. Note that the POSIX API consists of C functions which can be invoked from a C++ program.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00513*)

[RS_OSI_00104] The Operating System Interface shall support the reaction on process-external stimuli from devices. ⌈

Type:	valid
Description:	The Operating System shall enable processes to react on external stimuli from devices.
Rationale:	Application will react on reception of functional data, signals and timers from the platform. Certain computations shall be executed in reaction on these application-external stimuli.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00050*, *RS_Main_00060*, *RS_Main_00460*, *RS_Main_00160*)

[RS_OSI_00105] The Operating System Interface shall support the start of Execution Management. ⌈

Type:	valid
Description:	The Operating System shall provide means to start the Execution Management functional cluster as first process.
Rationale:	Execution Management is responsible for startup and shutdown of all processes of the AUTOSAR Adaptive Platform.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00049*, *RS_Main_00460*)

4.2.3 Operating System Requirements

[RS_OSI_00201] The Operating System shall provide mechanisms for system memory budgeting. ⌈

Type:	valid
Description:	The Operating System shall provide mechanisms to configure memory budgeting for each process or for groups of processes.
Rationale:	In order to ensure resource availability in the context of a multi-process system, the system integrator/architect may require a set of tools to configure memory budgeting for each process or for groups of process.
Dependencies:	–
Use Case:	security - protection against DoS attacks - resource starvation types.
Supporting Material:	–

⌋(*RS_Main_00002*, *RS_Main_00010*, *RS_Main_00011*, *RS_Main_00012*, *RS_Main_-00106*, *RS_Main_00150*, *RS_Main_00514*, *RS_SAF_21402*)

[RS_OSI_00202] The Operating System shall provide mechanisms for CPU time budgeting. ⌈

Type:	valid
Description:	The Operating System shall provide mechanisms to configure resource budgeting in terms of CPU time for each process or group of processes.
Rationale:	In order to ensure schedulability in the context of a multi-process system, the system integrator/architect may require a set of tools to configure CPU time allocated for each process or for groups of processes.
Dependencies:	–

▽

△

Use Case:	security - protection against DoS attacks - resource starvation types.
Supporting Material:	–

⌋(*RS_Main_00002*, *RS_Main_00010*, *RS_Main_00011*, *RS_Main_00012*, *RS_Main_00106*, *RS_Main_00150*, *RS_Main_00514*, *RS_SAF_21401*)

[RS_OSI_00203] The Operating System should provide mechanisms for binding processes to CPU cores. ⌈

Type:	valid
Description:	The `Operating System` should provide mechanisms for binding individual processes or groups of processes to CPU cores.
Rationale:	In order to ensure correct task schedulability, the system integrator may require a set of tools to configure the CPU affinity of processes. In a multi-core system, it may be relevant to ensure some process can only run on some CPU cores, to allow other less- or differently-restricted processes to concurrently progress.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00002*, *RS_Main_00010*, *RS_Main_00011*, *RS_Main_00012*, *RS_Main_00050*, *RS_Main_00106*, *RS_Main_00514*)

[RS_OSI_00204]{DRAFT} **The Operating System shall support authorized operating system object access for the software entities which are allowed to do so.** ⌈

Type:	draft
Description:	The `Operating System` shall provide access rights and permissions mechanisms to achieve secure data access and data exchange.
Rationale:	The `Operating System` consists of a collection of hardware and software objects, e.g. pipes, files. Safety or/and Security related requirements may be imposed to grant special access rights and permissions in order to avoid unauthorized access to communication channels or to ensure exclusive access to the process-specific data stored persistently.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00010*, *RS_Main_00170*, *RS_Main_00514*)

[RS_OSI_00206] The Operating System shall provide multi-process support for isolation of applications. ⌈

Type:	valid
Description:	The Operating System shall provide mechanisms to let multiple processes run isolated from each other.
Rationale:	Each process may have a different robustness, safety and security level. As a consequence, an incorrect memory access from one process execution shall not result in a corruption of memory in another process, unless the data area is explicitly shared. In addition, a process may not access or read data from another process without explicit data sharing.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00010*, *RS_Main_00049*, *RS_Main_00106*, *RS_SAF_21403*)

[RS_OSI_00207] The Operating System shall provide the capability to share code and data in an implicit manner. ⌈

Type:	valid
Description:	The Operating System shall provide mechanisms to run the same code and associated data copy in multiple processes by using a shared object mechanism, where data is either read-only or a process-private copy.
Rationale:	To allow more efficient memory usage both in runtime memory and non-volatile storage, as well as making platform updates faster, applications may be linked against shared code and data objects. Using shared objects is not intended however to be mandatory for use by the AUTOSAR Adaptive Platform, as security, safety or other constraints may require not to use this feature.
Dependencies:	–
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00410*, *RS_Main_00150*, *RS_Main_00503*)

[RS_OSI_00208]{DRAFT} **The Operating System shall only allow processes to access required functionality.** ⌈

Type:	draft
Description:	The Operating System shall ensure that either a process does not access operating system functionality via a system call at all, or the system call has to be authorized.
Rationale:	A process with full access to the Platform could cause significant damage.
Dependencies:	–
Use Case:	–

▽

△

Supporting Material:	This requirement was visible before 18-10 Specification Release as [RS_SEC_05009].

⌋*(RS_Main_00010, RS_Main_00514)*

4.3 Non-Functional Requirements

5 Requirements Tracing

The following tables reference the requirements specified in [7] 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_Main_00002]	AUTOSAR shall provide a software platform for high performance computing platforms	[RS_OSI_00100] [RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203]
[RS_Main_00010]	AUTOSAR shall support the development of safety related systems	[RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203] [RS_OSI_00204] [RS_OSI_00206] [RS_OSI_00208]
[RS_Main_00011]	AUTOSAR shall support the development of reliable systems	[RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203]
[RS_Main_00012]	AUTOSAR shall provide a software platform to support the development of highly available systems	[RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203]
[RS_Main_00026]	AUTOSAR shall support high speed and high bandwidth communication between executed SW	[RS_OSI_NA]
[RS_Main_00030]	AUTOSAR shall support development processes for safety related systems	[RS_OSI_NA]
[RS_Main_00049]	AUTOSAR shall provide an Execution Management for running multiple applications	[RS_OSI_00105] [RS_OSI_00206]
[RS_Main_00050]	AUTOSAR shall provide an Execution Framework towards applications to implement concurrent application internal control flows	[RS_OSI_00100] [RS_OSI_00104] [RS_OSI_00203]
[RS_Main_00060]	AUTOSAR shall provide a standardized software interface for communication between Applications	[RS_OSI_00104]

Requirement	Description	Satisfied by
[RS_Main_00080]	AUTOSAR shall provide means to describe a component model for Application Software	[RS_OSI_NA]
[RS_Main_00106]	AUTOSAR shall provide the possibility to extend the software with new SWCs without recompiling the platform foundation	[RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203] [RS_OSI_00206]
[RS_Main_00140]	AUTOSAR shall provide network independent communication mechanisms for applications	[RS_OSI_NA]
[RS_Main_00150]	AUTOSAR shall support the deployment and reallocation of AUTOSAR Application Software	[RS_OSI_00100] [RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00207]
[RS_Main_00160]	AUTOSAR shall provide means to describe interfaces of the entire system	[RS_OSI_00104] [RS_OSI_NA]
[RS_Main_00161]	AUTOSAR shall provide a unified way to describe software systems deployed to Adaptive and / or Classic platforms	[RS_OSI_NA]
[RS_Main_00170]	AUTOSAR shall provide secure access to ECU data and services	[RS_OSI_00204]
[RS_Main_00180]	AUTOSAR shall provide mechanisms to protect intellectual property in a shared development process	[RS_OSI_NA]
[RS_Main_00190]	AUTOSAR shall support standardized interoperability with non-AUTOSAR software	[RS_OSI_NA]
[RS_Main_00230]	AUTOSAR shall support network topologies including gateways	[RS_OSI_NA]
[RS_Main_00250]	AUTOSAR methodology shall provide a predefinition of typical roles and activities	[RS_OSI_NA]
[RS_Main_00260]	AUTOSAR shall provide diagnostics means during runtime, for production and services purposes	[RS_OSI_NA]
[RS_Main_00261]	AUTOSAR shall provide means for calibration	[RS_OSI_NA]
[RS_Main_00270]	AUTOSAR shall provide mitigation strategies towards new releases	[RS_OSI_NA]
[RS_Main_00280]	AUTOSAR shall support standardized automotive communication protocols	[RS_OSI_NA]
[RS_Main_00300]	AUTOSAR shall provide data exchange formats to support work-share in large inter and intra company development groups	[RS_OSI_NA]

Requirement	Description	Satisfied by
[RS_Main_00301]	AUTOSAR shall specify profiles for data exchange to support work-share in large inter- and intra-company development groups	[RS_OSI_NA]
[RS_Main_00310]	AUTOSAR shall support hierarchical Application Software design methods	[RS_OSI_NA]
[RS_Main_00320]	AUTOSAR shall provide formats to specify system development	[RS_OSI_NA]
[RS_Main_00340]	AUTOSAR shall support the continuous timing requirement analysis	[RS_OSI_NA]
[RS_Main_00350]	AUTOSAR specifications shall be analyzable and support according methods to demonstrate the achievement of safety related properties	[RS_OSI_NA]
[RS_Main_00360]	AUTOSAR shall support variant management	[RS_OSI_NA]
[RS_Main_00410]	AUTOSAR shall provide specifications for routines commonly used by Application Software to support sharing and optimization	[RS_OSI_00207]
[RS_Main_00420]	AUTOSAR shall use established software standards and consolidate de-facto standards for basic software functionality	[RS_OSI_00100]
[RS_Main_00440]	AUTOSAR shall standardize access to non-volatile memory	[RS_OSI_NA]
[RS_Main_00445]	AUTOSAR shall standardize access to crypto-specific HW and SW	[RS_OSI_NA]
[RS_Main_00460]	AUTOSAR shall standardize methods to organize mode management on Application, ECU and System level	[RS_OSI_00104] [RS_OSI_00105]
[RS_Main_00480]	AUTOSAR shall support the test of implementations	[RS_OSI_NA]
[RS_Main_00490]	AUTOSAR processes shall be compliant to ISO26262	[RS_OSI_NA]
[RS_Main_00491]	AUTOSAR shall provide means for logging	[RS_OSI_NA]
[RS_Main_00500]	AUTOSAR shall provide naming conventions	[RS_OSI_NA]
[RS_Main_00501]	AUTOSAR shall support redundancy concepts	[RS_OSI_NA]
[RS_Main_00503]	AUTOSAR shall support change of communication and application software at runtime.	[RS_OSI_00207]

Requirement	Description	Satisfied by
[RS_Main_00507]	AUTOSAR shall reflect the stages of a software system development in a formal model description	[RS_OSI_NA]
[RS_Main_00510]	AUTOSAR shall support secure onboard communication	[RS_OSI_NA]
[RS_Main_00511]	AUTOSAR shall support virtualization	[RS_OSI_NA]
[RS_Main_00513]	AUTOSAR shall support language bindings for different programming languages	[RS_OSI_00103]
[RS_Main_00514]	AUTOSAR shall support the development of secure systems	[RS_OSI_00201] [RS_OSI_00202] [RS_OSI_00203] [RS_OSI_00204] [RS_OSI_00208]
[RS_Main_00650]	AUTOSAR shall support up - and download of data and software	[RS_OSI_NA]
[RS_Main_00652]	AUTOSAR shall support the translation between signal-based and service-oriented communication	[RS_OSI_NA]
[RS_Main_00653]	AUTOSAR shall provide an abstract description of the vehicle VFB communications independent of platform	[RS_OSI_NA]
[RS_Main_01001]	AUTOSAR shall support intra ECU communication	[RS_OSI_NA]
[RS_Main_01002]	AUTOSAR shall support service-oriented communication	[RS_OSI_NA]
[RS_Main_01003]	AUTOSAR shall support data-oriented communication	[RS_OSI_NA]
[RS_Main_01004]	AUTOSAR shall support standards for wireless off-board communication	[RS_OSI_NA]
[RS_Main_01005]	AUTOSAR shall establish communication paths dynamically	[RS_OSI_NA]
[RS_Main_01007]	AUTOSAR communication shall assure quality of service on communication	[RS_OSI_NA]
[RS_Main_01008]	AUTOSAR shall provide secure communication with off-board entities	[RS_OSI_NA]
[RS_Main_01025]	AUTOSAR shall support debugging of software on the target and onboard	[RS_OSI_NA]
[RS_Main_01026]	AUTOSAR shall support tracing and profiling on the target and onboard	[RS_OSI_NA]

Requirement	Description	Satisfied by
[RS_SAF_21401]	The OS shall support a mechanism that prevents starvation of applications or processes on the basis of CPU usage (under the respect of available resources).	[RS_OSI_00202]
[RS_SAF_21402]	The OS shall support resource reservation for memory in the interval [min,max]. If max is not specified it shall be considered as unlimited.	[RS_OSI_00201]
[RS_SAF_21403]	Operating System shall ensure that only allowed memory accesses are made.	[RS_OSI_00206]

6 References

[1] Safety Requirements for AUTOSAR Adaptive Platform and AUTOSAR Classic Platform
AUTOSAR_RS_Safety

[2] Requirements on Execution Management
AUTOSAR_RS_ExecutionManagement

[3] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[4] Glossary
AUTOSAR_TR_Glossary

[5] IEEE Standard for Information Technology- Standardized Application Environment Profile (AEP)-POSIX Realtime and Embedded Application Support
https://standards.ieee.org/findstds/standard/1003.13-2003.html

[6] Standard for Information Technology–Portable Operating System Interface (POSIX(R)) Base Specifications, Issue 7
http://pubs.opengroup.org/onlinepubs/9699919799/

[7] Main Requirements
AUTOSAR_RS_Main