

Document Title	Requirements on Persistency
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	857

Document Status	published
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	R21-11

Document Change History			
Date	Release	Changed by	Description
2021-11-25	R21-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Re-introduced support of finalization of persistent data • Removed support of removal of persistent data • Harmonized names between SWS and RS • Improved formal aspects of requirements
2020-11-30	R20-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Added support for file meta-data • Added up-traces to RS Safety • Adapted to common document structure • Added requirement history
2019-11-28	R19-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Statistic data • Initialization/De-initialization • Changed Document Status from Final to published
2019-03-29	19-03	AUTOSAR Release Management	<ul style="list-style-type: none"> • Updated introduction • Information on wear leveling • Finalization was removed
2018-10-31	18-10	AUTOSAR Release Management	<ul style="list-style-type: none"> • Restructured document • Added resource handling • Added UCM related requirements
2018-03-29	18-03	AUTOSAR Release Management	<ul style="list-style-type: none"> • Added requirement on configuration • Removed requirement on authorization

2017-10-27	17-10	AUTOSAR Release Management	<ul style="list-style-type: none">• Requirements on Secure Storage• Requirements on Safe Storage
2017-03-31	17-03	AUTOSAR Release Management	<ul style="list-style-type: none">• 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 Document	6
2	Conventions to be Used	7
2.1	Requirements Guidelines	7
3	Acronyms and Abbreviations	8
4	Requirements Specification	9
4.1	Functional Overview	9
4.2	Functional Requirements	9
4.2.1	Configuration of Persistent Data	9
4.2.2	Storage of Persistent Data	10
4.2.3	Secure Storage of Persistent Data	12
4.2.4	Safe Storage of Persistent Data	12
4.2.5	Installation, Update, and Roll-back of Persistent Data	13
4.2.6	Resource handling for Persistent Data	14
4.3	Non-Functional Requirements	15
5	Requirements Tracing	16
6	References	17
A	History of Requirements	18
A.1	Requirement History of this Document According to AUTOSAR Release 17-03	18
A.1.1	Added Traceables in 17-03	18
A.1.2	Changed Traceables in 17-03	18
A.1.3	Deleted Traceables in 17-03	18
A.2	Requirement History of this Document According to AUTOSAR Release 17-10	18
A.2.1	Added Traceables in 17-10	18
A.2.2	Changed Traceables in 17-10	18
A.2.3	Deleted Traceables in 17-10	18
A.3	Requirement History of this Document According to AUTOSAR Release 18-03	19
A.3.1	Added Traceables in 18-03	19
A.3.2	Changed Traceables in 18-03	19
A.3.3	Deleted Traceables in 18-03	19
A.4	Requirement History of this Document According to AUTOSAR Release 18-10	19
A.4.1	Added Traceables in 18-10	19
A.4.2	Changed Traceables in 18-10	19
A.4.3	Deleted Traceables in 18-10	19
A.5	Requirement History of this Document According to AUTOSAR Release 19-03	19

A.5.1	Added Traceables in 19-03	19
A.5.2	Changed Traceables in 19-03	20
A.5.3	Deleted Traceables in 19-03	20
A.6	Requirement History of this Document According to AUTOSAR Release R19-11	20
A.6.1	Added Traceables in R19-11	20
A.6.2	Changed Traceables in R19-11	20
A.6.3	Deleted Traceables in R19-11	20
A.7	Requirement History of this Document According to AUTOSAR Release R20-11	20
A.7.1	Added Traceables in R20-11	20
A.7.2	Changed Traceables in R20-11	20
A.7.3	Deleted Traceables in R20-11	20
A.8	Requirement History of this Document According to AUTOSAR Release R21-11	21
A.8.1	Added Traceables in R21-11	21
A.8.2	Changed Traceables in R21-11	21
A.8.3	Deleted Traceables in R21-11	21

1 Scope of Document

This document specifies the requirements of Adaptive Applications to the functional cluster Persistency of the AUTOSAR Adaptive Platform. The motivation is to provide a standardized and portable way to store and write data persistently.

2 Conventions to be Used

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

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

2.1 Requirements Guidelines

There are no special guidelines for requirements for the functional cluster Persistency.

3 Acronyms and Abbreviations

There are no acronyms and abbreviations relevant within this document that are not included in the AUTOSAR Glossary [\[2\]](#).

4 Requirements Specification

This chapter describes all requirements driving the work to define the functional cluster Persistency.

The functional cluster Persistency will be referenced as “Persistency” in the remainder of this document.

4.1 Functional Overview

The AUTOSAR Adaptive Platform Persistency provides services for Adaptive Applications and other functional clusters of the AUTOSAR Adaptive Platform. The AUTOSAR Adaptive Platform Persistency is responsible for all aspects which regard the storage/retrieval of data, and therefore it has to deal with:

- Persistently storing data over boot and ignition cycles.
- Accessing data which has been stored persistently.
- Using a unique identifier to access data.
- Reading and writing data from/to files.
- Encryption of persistent data.
- Error detection and correction of stored data.
- Monitoring storage space.

4.2 Functional Requirements

The Persistency shall fulfill the following functional requirements.

4.2.1 Configuration of Persistent Data

[RS_PER_00010]{DRAFT} The layout of persistent data shall be configurable [

Description:	Persistency shall support configuration of provided key-value storage and file storage.
Rationale:	Generation of interfaces
Dependencies:	–
Use Case:	An Adaptive Application or a functional cluster needs access to persistent data and expects a dedicated interface for each set of data.



△

Supporting Material:	–
-----------------------------	---

](RS_Main_00440)

4.2.2 Storage of Persistent Data

[RS_PER_00001]{DRAFT} Persistency shall support storage of persistent data [

Description:	Persistency shall support persistent storage of data of an Adaptive Application. In case of direct storage to flash memory or other storage hardware that has a limited number of write cycles, the implementation of Persistency shall take care of wear leveling.
Rationale:	Applications need to preserve data from one run-time to the next.
Dependencies:	–
Use Case:	Applications have data like settings, diagnostic data, calibration data, or error logs that they want to store on a file system or in a database.
Supporting Material:	–

](RS_Main_00440)

[RS_PER_00018]{DRAFT} Persistency shall support central initialization and shutdown [

Description:	By using a central mechanism, an application developer shall be able to control the life cycle of Persistency together with the life cycle of other functional clusters with direct interfaces to the application.
Rationale:	Preparation and cleanup of the in-memory structures of Persistency is required to ensure a reliable life cycle handling.
Dependencies:	–
Use Case:	The manifest configuration of Persistency needs to be parsed before any storage can be accessed, and before shutdown, all data of Persistency should be persisted and the structures cleaned up.
Supporting Material:	–

](RS_Main_00011)

[RS_PER_00002]{DRAFT} Persistency shall support to retrieve data that has been persistently stored on a platform instance [

Description:	Persistency shall provide the functionality to load data which is persistently stored.
Rationale:	Load of persistently stored data
Dependencies:	–
Use Case:	An Adaptive Application or functional cluster which stores persistent data needs to restore it after a restart of the Adaptive Application or the platform.
Supporting Material:	–

](RS_Main_00440)

[RS_PER_00003]{DRAFT} Persistency shall support identification of data using a unique identifier [

Description:	Data shall be stored in way that it can be accessed from an Adaptive Application or a functional cluster by using a unique identifier e.g. identify a value by a key.
Rationale:	Load of persistently stored data
Dependencies:	–
Use Case:	Storage of a variety of different data objects that can be accessed individually for loading.
Supporting Material:	–

](RS_Main_00440)

[RS_PER_00004]{DRAFT} Persistency shall support access to file-like structures [

Description:	Persistency shall provide a standardized way to access file-like structures. Adaptive Applications and the other functional clusters shall be able to read and write data from file-like structures, and read associated meta data (e.g. access time). Persistent data can be represented in multiple ways, e.g. human-readable format or binary. Every format of data needs to be accessible by Persistency.
Rationale:	Persistency shall emulate the basic features of a file system, because PSE51 does not contain file system support.
Dependencies:	–
Use Case:	Store information that is not structured as key-value pairs.
Supporting Material:	–

](RS_Main_00440)

4.2.3 Secure Storage of Persistent Data

[RS_PER_00005]{DRAFT} Persistency shall support encryption/decryption of persistent data [

Description:	Persistency shall provide a standardized way to encrypt/decrypt persistent data.
Rationale:	Support of data encryption
Dependencies:	–
Use Case:	Storage of persistent data that shall be encrypted for security reasons.
Supporting Material:	–

]([RS_Main_00514](#))

4.2.4 Safe Storage of Persistent Data

[RS_PER_00008]{DRAFT} Persistency shall support detection of data corruption in persistent memory [

Description:	Persistency shall support detection of data corruption in persistently stored data. The corruption may be caused by systematic or random failures. To be able to detect corrupted data, some redundancy is needed, which can be anything from a checksum to a full copy. The actual mechanisms and the granularity of redundancy are subject to configuration.
Rationale:	Applications need to be sure to read valid data.
Dependencies:	–
Use Case:	Notification to an Adaptive Application or functional cluster in case of corrupted data in persistent memory, which is essential for safety use cases. The detection of data corruption is also necessary to support data recovery mechanisms.
Supporting Material:	–

]([RS_Main_00011](#), [RS_SAF_21501](#), [RS_SAF_21502](#))

[RS_PER_00009]{DRAFT} Persistency shall support data recovery mechanisms if persistent data was corrupted [

Description:	Persistency shall support a recovery mechanism if corruption of persistently stored data was detected. To be able to recover corrupted data, a redundant copy of the data is needed. The actual mechanisms and the granularity of redundancy are subject to configuration. Persistency shall also support a notification of the application in case recovery took place.
Rationale:	Applications want to recover corrupted data.





Dependencies:	–
Use Case:	If corruption of persistent data was detected it shall be possible to recover corrupted data.
Supporting Material:	–

]([RS_Main_00011](#), [RS_SAF_21501](#))

4.2.5 Installation, Update, and Roll-back of Persistent Data

[RS_PER_00012]{DRAFT} Persistency shall support installation of persistent data [

Description:	Persistency shall allow for installation of pre-configured values in key-value storages and pre-configured files in a file storage. The pre-configured data is provided by the manifest.
Rationale:	It shall be possible to install an application with a preset.
Dependencies:	–
Use Case:	Providing initial or fixed content for key-value storages and file storages.
Supporting Material:	–

]([RS_Main_00150](#), [RS_Main_00503](#))

[RS_PER_00013]{DRAFT} Persistency shall support update of persistent data [

Description:	Persistency shall allow for an update of values in key-value storages and of files in a file storage. The update strategy and updated data is provided by the manifest.
Rationale:	It shall be possible to update an application and set a new preset.
Dependencies:	–
Use Case:	Providing updated content for key-value storages and file storages.
Supporting Material:	–

]([RS_Main_00150](#), [RS_Main_00503](#))

[RS_PER_00014]{DRAFT} Persistency shall support roll-back of persistent data [

Description:	Persistency shall allow for a roll-back of values in key-value storages and files in a file storage to the state before an update.
---------------------	--





Rationale:	It shall be possible to roll back an application and return persisted data to its previous state.
Dependencies:	–
Use Case:	Reverting the content of key-value storages and file storages.
Supporting Material:	–

]([RS_Main_00150](#), [RS_Main_00503](#))

[RS_PER_00016]{DRAFT} Persistency shall support finalization of an update of persistent data [

Description:	Persistency shall allow for a finalization of an update of values in key-value storages and files in a file storage.
Rationale:	It shall be possible to finalize an update of an application and its persisted data.
Dependencies:	–
Use Case:	Finalizing the update of key-value storages and file storages.
Supporting Material:	–

]([RS_Main_00150](#), [RS_Main_00503](#))

4.2.6 Resource handling for Persistent Data

[RS_PER_00011]{DRAFT} Persistency shall be able to ensure and limit the amount of storage used by persisted data [

Description:	Persistency shall support monitoring of the storage space allocated by persistently stored data. It shall ensure that a configurable amount of storage space is always available for stored data, and that the stored data never surpasses a configurable limit.
Rationale:	Avoid situations where applications cannot run reliably because they cannot access the required amount of storage, or because another application uses too much storage.
Dependencies:	–
Use Case:	Ensuring reliability of the access to the persistently stored data of a single process, and ensuring overall reliability of applications regarding access to persistently stored data.
Supporting Material:	–

]([RS_Main_00011](#))

[RS_PER_00017]{DRAFT} Persistency shall be able to report the amount of currently used storage [

Description:	Persistency shall support querying the amount of storage currently allocated by persisted data.
Rationale:	It shall be possible to acquire information about persistent storage.
Dependencies:	–
Use Case:	Polling of the current size of persisted data using a diagnostic service.
Supporting Material:	–

|(RS_Main_00440)

4.3 Non-Functional Requirements

The Persistency currently does not have any non-functional requirements.

5 Requirements Tracing

The following table references the requirements specified in the AUTOSAR Main Requirements [3] and the AUTOSAR Safety Requirements [4], and links to the fulfillments of these.

Requirement	Description	Satisfied by
[RS_Main_00011]	Mechanisms for Reliable Systems	[RS_PER_00008] [RS_PER_00009] [RS_PER_00011] [RS_PER_00018]
[RS_Main_00150]	AUTOSAR shall support the deployment and reallocation of AUTOSAR Application Software	[RS_PER_00012] [RS_PER_00013] [RS_PER_00014] [RS_PER_00016]
[RS_Main_00440]	AUTOSAR shall standardize access to non-volatile memory	[RS_PER_00001] [RS_PER_00002] [RS_PER_00003] [RS_PER_00004] [RS_PER_00010] [RS_PER_00017]
[RS_Main_00503]	AUTOSAR shall support change of communication and application software at runtime.	[RS_PER_00012] [RS_PER_00013] [RS_PER_00014] [RS_PER_00016]
[RS_Main_00514]	System Security Support	[RS_PER_00005]
[RS_SAF_21501]	Persistency shall add integrity information to the persistent data if such a mechanism does not already exist in the operating system.	[RS_PER_00008] [RS_PER_00009]
[RS_SAF_21502]	Persistency shall check the integrity of persistent data when reading it if this is not already done by the operating system.	[RS_PER_00008]

6 References

- [1] Standardization Template
AUTOSAR_TPS_StandardizationTemplate
- [2] Glossary
AUTOSAR_TR_Glossary
- [3] Main Requirements
AUTOSAR_RS_Main
- [4] Safety Requirements for AUTOSAR Adaptive Platform and AUTOSAR Classic Platform
AUTOSAR_RS_Safety

A History of Requirements

Please note that the lists in this chapter also include requirements that have been removed from the specification in a later version. These requirements do not appear as hyperlinks in the document.

A.1 Requirement History of this Document According to AUTOSAR Release 17-03

A.1.1 Added Traceables in 17-03

[\[RS_PER_00001\]](#) [\[RS_PER_00002\]](#) [\[RS_PER_00003\]](#) [\[RS_PER_00004\]](#)

A.1.2 Changed Traceables in 17-03

none

A.1.3 Deleted Traceables in 17-03

none

A.2 Requirement History of this Document According to AUTOSAR Release 17-10

A.2.1 Added Traceables in 17-10

[\[RS_PER_00005\]](#) [\[RS_PER_00007\]](#) [\[RS_PER_00008\]](#) [\[RS_PER_00009\]](#)

A.2.2 Changed Traceables in 17-10

[\[RS_PER_00001\]](#) [\[RS_PER_00002\]](#) [\[RS_PER_00003\]](#) [\[RS_PER_00004\]](#)

A.2.3 Deleted Traceables in 17-10

none

A.3 Requirement History of this Document According to AUTOSAR Release 18-03

A.3.1 Added Traceables in 18-03

[\[RS_PER_00010\]](#)

A.3.2 Changed Traceables in 18-03

[\[RS_PER_00002\]](#) [\[RS_PER_00003\]](#) [\[RS_PER_00004\]](#) [\[RS_PER_00008\]](#)

A.3.3 Deleted Traceables in 18-03

[\[RS_PER_00007\]](#)

A.4 Requirement History of this Document According to AUTOSAR Release 18-10

A.4.1 Added Traceables in 18-10

[\[RS_PER_00011\]](#) [\[RS_PER_00012\]](#) [\[RS_PER_00013\]](#) [\[RS_PER_00014\]](#) [\[RS_PER_00015\]](#) [\[RS_PER_00016\]](#)

A.4.2 Changed Traceables in 18-10

none

A.4.3 Deleted Traceables in 18-10

none

A.5 Requirement History of this Document According to AUTOSAR Release 19-03

A.5.1 Added Traceables in 19-03

none

A.5.2 Changed Traceables in 19-03

[\[RS_PER_00001\]](#)

A.5.3 Deleted Traceables in 19-03

[\[RS_PER_00016\]](#)

A.6 Requirement History of this Document According to AUTOSAR Release R19-11

A.6.1 Added Traceables in R19-11

[\[RS_PER_00017\]](#) [\[RS_PER_00018\]](#)

A.6.2 Changed Traceables in R19-11

none

A.6.3 Deleted Traceables in R19-11

none

A.7 Requirement History of this Document According to AUTOSAR Release R20-11

A.7.1 Added Traceables in R20-11

none

A.7.2 Changed Traceables in R20-11

[\[RS_PER_00004\]](#)

A.7.3 Deleted Traceables in R20-11

none

A.8 Requirement History of this Document According to AUTOSAR Release R21-11

A.8.1 Added Traceables in R21-11

[RS_PER_00016]

A.8.2 Changed Traceables in R21-11

[RS_PER_00001] [RS_PER_00004] [RS_PER_00008] [RS_PER_00009] [RS_PER_-00012] [RS_PER_00013]

A.8.3 Deleted Traceables in R21-11

[RS_PER_00015]