Document Title	Requirements on Transformer
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	657

Document Status	Final
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	4.4.0

Document Change History

Date	Release	Changed by	Description
2018-10-31	4.4.0	AUTOSAR Release Management	• Extended Serialization for Data Structures in SOME/IP with tag/length/value encoding • Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2017-12-08	4.3.1	AUTOSAR Release Management	Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2016-11-30	4.3.0	AUTOSAR Release Management	Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2015-07-31	4.2.2	AUTOSAR Release Management	Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2014-10-31	4.2.1	AUTOSAR Release Management	Initial Release

Disclaimer

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 this document is to define a common set of basic requirements that apply to all Transformers in AUTOSAR.

For the sake of simplicity all requirements on Transformers are maintained in this one document.

The functional requirements defined in this document shall be referenced in each Software Specification (SWS) document of the specific Transformers.

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 table references the features specified in [3] and links to the fulfillments of these.

Feature	Description	Satisfied by
[RS_BRF_01056]	AUTOSAR BSW modules shall provide standardized interfaces	[SRS_Xfrm_00002]
[RS_BRF_01316]	AUTOSAR RTE shall support data transformation transparent to the Software Components	[SRS_Xfrm_00001] [SRS_Xfrm_00002] [SRS_Xfrm_00003] [SRS_Xfrm_00004] [SRS_Xfrm_00005] [SRS_Xfrm_00006] [SRS_Xfrm_00007] [SRS_Xfrm_00008] [SRS_Xfrm_00009] [SRS_Xfrm_00010] [SRS_Xfrm_00011] [SRS_Xfrm_00101] [SRS_Xfrm_00102] [SRS_Xfrm_00103] [SRS_Xfrm_00105] [SRS_Xfrm_00201] [SRS_Xfrm_00202]
[RS_BRF_01424]	AUTOSAR services shall support communication services	[SRS_Xfrm_00001]
[RS_BRF_01544]	AUTOSAR communication shall define transmission and reception of communication data	[SRS_Xfrm_00201] [SRS_Xfrm_00202]
[RS_BRF_01560]	AUTOSAR communication shall support mapping of signals into transferrable protocol data units	[SRS_Xfrm_00201] [SRS_Xfrm_00202]
[RS_BRF_01592]	AUTOSAR communication shall offer data transfer on user request, time based, and requested via the underlying bus	[SRS_Xfrm_00201] [SRS_Xfrm_00202]
[RS_Main_00280]	AUTOSAR shall support standardized automotive communication protocols	[SRS_Xfrm_00106]

4 Requirements

4.1 General Requirements on Transformers

A transformer takes data from the RTE, works on them and returns the output back to the RTE. It can both serialize/linearize data (transform them from a structured into a linear form) and transform (modify or extend linear data) them (e.g add a checksum).

Transformers are BSW modules in the System Service Cluster which provide services to the RTE. The transformers are executed by the RTE when the RTE needs the service which a transformer provides.

A transformer is no library because transformers can hold an internal state but they can work as well stateless.

It is possible to connect a set of transformers together into a transformer chain. The RTE coordinates the execution of the transformer chain and calls the transformers of the chain exactly in the specified order. Using that mechanism, inter-ECU communication is transformed if configured accordingly. This configuration is done in the [4, System Template]. The maximum length of a transformer chain is limited to 255 transformers.

The order of transformers configured in the [4, System Template] represents the order on the sending side. The order on the receiving side is the inverse of the sending side.

[SRS_Xfrm_00001] A transformer shall work on data given by the Rte ⌈

Type:	valid
Description:	A transformer shall work on data given by the Rte and outputs the result back to the Rte.
Rationale:	A transformer is called by the Rte with input data, it works in the data and produces a result. This result is given back to the Rte.
Dependencies:	–
Use Case:	A transformer takes a complex data element, serializes it and gives a linear byte array back to the Rte. The Rte hands those data over to the Com stack for inter-ECU communication. A transformer calculates a checksum over the data and appends it as header or footer.
Supporting Material:	–

⌋*(RS_BRF_01316, RS_BRF_01424)*

[SRS_Xfrm_00002] A transformer shall provide fixed interfaces ⌈

Type:	valid
Description:	A transformer shall provide fixed interfaces to the Rte which uses those interfaces to call the transformer.
Rationale:	A transformer needs to provide certain interfaces to the Rte to enable the Rte to call the transformer without further special information. The signature of the interfaces depend on the position of the transformer in the transformer chain and the data element to transform.
Dependencies:	[SRS_Xfrm_00001]
Use Case:	A transformer which provides a certain transformation functionality can be used in multiple systems if the input, output data and functional requirements are the same on the systems.
Supporting Material:	–

⌋*(RS_BRF_01316, RS_BRF_01056)*

[SRS_Xfrm_00003] A Transformer shall support in-place and copy buffering ⌈

Type:	valid
Description:	A transformer shall support in-place and copy buffer mechanisms which allow a configuration of where to store the result of the transformer.
Rationale:	Depending on the transformer's functionality (e.g. serialization of complex data or adding checksums) different buffer mechanisms are more efficient than others.
Dependencies:	–
Use Case:	Serialization of complex data needs an input and an output buffer. Appending a checksum works efficient inside the same buffer.
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00004] A transformer shall support error handling ⌈

Type:	valid
Description:	A transformer shall return its errors to the RTE.
Rationale:	The RTE as coordinator of the execution of transformers has to be informed about error which occurred during the execution of the transformers.
Dependencies:	[SRS_Xfrm_00001]
Use Case:	The RTE aborts the execution of a transformer chain after an error. The RTE notifies SWCs about transformer errors which lead to no abortion of a transformer chain.
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00005] A transformer shall be able to deal with more data than expected ⌈

Type:	valid
Description:	A transformer shall be able to deal with longer input data than it expects. A transformer shall discard the unexpected data but shall tolerate the expected fraction.
Rationale:	When an existing ECU is used in newer systems, the newer systems might need to transport additional data. These can be appended to the end of the already existing data on the network. The existing ECU has to be able to cope with that and ignore new added part at the end.
Dependencies:	–
Use Case:	Extension of data on the bus by appending additional elements.
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00006] A Transformer shall support concurrent execution ⌈

Type:	valid
Description:	A transformer shall support concurrent execution.
Rationale:	The RTE coordinates the execution of the transformers. If the same data elements is access concurrently (e.g. by two runnables) or the same transformer is used in more than one transformer chain, the RTE might execute it concurrently.
Dependencies:	[SRS_Xfrm_00001]
Use Case:	Usage of a transformer in more than one transformer chain. Access to the same data element (of the same port) from two runnables.
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00007] A deserializer transformer shall support extraction of data ⌈

Type:	valid
Description:	A transformer which resides on the receiver side and outputs an ImplementationDataType to the RTE shall support the output of a data element which contains only a subset of the data which are contained in the input.
Rationale:	A transformer which resides on the receiver side and implements the deserialization functionality of a serializer transformer shall be able to output a complex data structure which consists only of a subset of the data structure which was used as input to the serializer transformer that produced the data which are subject to retransformation.

▽

△

Dependencies:	–
Use Case:	PortInterfaceMapping where the receiver only reads a subset of the data structure sent by the sender.
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00008] A transformer shall specify its output format ⌈

Type:	valid
Description:	A transformer shall clearly specify how the format of its output data looks like.
Rationale:	The output of a transformer shall be clearly specified to enable the implementation of the corresponding transformer on the remote ECU to be able to work with the data.
Dependencies:	–
Use Case:	Transformer on sender and receiver ECU are implemented by different parties/vendors. Processor architecture of sender and receiver ECU are different (e.g. little endian and big endian).
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00009] A fixed set of transformer classes shall exist ⌈

Type:	valid
Description:	A fixed set of transformer classes shall exist in AUTOSAR.
Rationale:	This gives the possibility to group transformers with the same functionality and define common properties and functionality for them.
Dependencies:	–
Use Case:	Common functionality Common abstract error set
Supporting Material:	Further Supporting Material

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00010] Each transformer class shall provide a fixed set of abstract errors ⌈

Type:	valid
Description:	Each transformer class shall provide a fixed set of abstract errors.
Rationale:	Each transformer of a class can return the errors defined for that class. This enables the SWCs to check for the existence for abstracted transformer errors without the knowledge whether a transformer is used at all and if yes what transformer is used.
Dependencies:	[SRS_Xfrm_00009]
Use Case:	Abstraction of used transformers for the SWCs
Supporting Material:	Further Supporting Material

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00011] A transformer shall belong to a specific transformer class ⌈

Type:	valid
Description:	A transformer shall belong to a specific transformer class.
Rationale:	This specifies which functionality and which errors a transformer has to provide.
Dependencies:	[SRS_Xfrm_00009]
Use Case:	Abstraction of used transformers for the SWCs
Supporting Material:	Further Supporting Material

⌋*(RS_BRF_01316)*

4.2 SOME/IP Transformer Requirements

The **Scalable service-Oriented MiddlewarE over IP (SOME/IP) Transformer** linearizes data with the SOME/IP on-the-wire format and specifies an automotive/embedded RPC mechanism for inter-ECU Client/Server communication.

The basic motivation to specify "yet another RPC-Mechanism" instead of using an existing infrastructure/technology is the goal to have a technology that:

- Fulfills the hard requirements regarding resource consumption in an embedded world
- Is compatible through as many use-cases and communication partners as possible
- Provides the features required by automotive use-cases
- Is scalable from tiny to large platforms
- Can be implemented on different operating system (i.e. AUTOSAR, GENIVI, and OSEK) and even embedded devices without operating system

[SRS_Xfrm_00101] The SOME/IP Transformer shall define the serialization of atomic and structured data elements into linear arrays ⌈

Type:	valid
Description:	The SOME/IP Transformer shall define the serialization of atomic and structured data elements into linear arrays.
Rationale:	An algorithm to linearize the data is necessary to enable communication if SOME/IP transformers of different vendors/implementors are used in the different ECU which communicate with each other.
Dependencies:	–
Use Case:	Inter-ECU Sender/Receiver communication Inter-ECU Client/Server communication
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00102] The SOME/IP Transformer shall define a protocol for inter-ECU Client/Server communication ⌈

Type:	valid
Description:	The SOME/IP Transformer shall define a protocol for inter-ECU Client/Server communication.
Rationale:	A protocol is necessary to meet the functionality and requirements of the RTE and the SWCs regarding Client/Server communication.

▽

△

Dependencies:	–
Use Case:	Inter-ECU Client/Server communication
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00103] The SOME/IP Transformer shall support exception notification of applications ⌈

Type:	valid
Description:	The SOME/IP Transformer shall support exception notification of applications
Rationale:	Non-AUTOSR ECUs can throw application exceptions which can be transported by SOME/IP. Access whether such an exception occurred shall be supported in AUTOSAR's SWCs.
Dependencies:	–
Use Case:	Communication with non-AUTOSAR ECUs
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00105] The SOME/IP Transformer shall support autonomous error reactions on the server side for client/server communication ⌈

Type:	valid
Description:	The SOME/IP Transformer shall support autonomous error reactions on the server side for client/server communication
Rationale:	A lot of error reactions on requests for client/server communication don't need active contribution of applications. For those the error handling shall be transparent which means that the error messages are generated directly by the BSW.
Dependencies:	–
Use Case:	Transparent Error Handling
Supporting Material:	–

⌋*(RS_BRF_01316)*

[SRS_Xfrm_00106]{DRAFT} **The SOME/IP Transformer shall support serialization of extensible data structs and methods ⌈**

Type:	valid
Description:	The SOME/IP Transformer shall support serialization of structs and methods with tagged members/arguments. These tags serve as identifier for the individual member/argument. This allows that during deserialization unknown members/arguments can be skipped and that optional members which are not available do not need to be serialized.
Rationale:	This mechanism is required to realize optional members. Moreover, it allows easy extensibility of an interface definition while maintaining backward/forward compatibility.
Dependencies:	–
Use Case:	A struct/method in an interface can be extended by members/arguments at arbitrary positions at sender side. A receiver does not need to be adapted if it doesn't need the new members/arguments. A sender does not need to send struct members which are optional and currently not available.
Supporting Material:	–

⌋*(RS_Main_00280)*

4.3 COM Based Transformer Requirements

The transformer feature provides functionality to chain arbitrary transformers when sending and receiving data in the RTE. This functionality shall also be available when the target bus system uses a fixed communication matrix with packed data representations.

[SRS_Xfrm_00201] The COM Based Transformer shall define the serialization of atomic and structured data elements into linear arrays based on a fixed data mapping ⌈

Type:	valid
Description:	The COM Based Transformer shall define the serialization of atomic and structured data elements into linear arrays based on a fixed data mapping.
Rationale:	In order to support the interaction with a statically defined communication matrix the COM Based Transformer takes the information how to serialize the data from the fixed data mapping of the System Description.
Dependencies:	–
Use Case:	Inter-ECU Sender/Receiver communication based on fixed communication matrix
Supporting Material:	–

⌋*(RS_BRF_01316, RS_BRF_01544, RS_BRF_01560, RS_BRF_01592)*

[SRS_Xfrm_00202] The COM Based Transformer shall take its configuration from the COM module ⌈

Type:	valid
Description:	The COM Based Transformer shall take its configuration from the COM module[5].
Rationale:	The COM module already copes with the handling of the communication matrix. The information is taken from the System Description and placed in the COM module's Ecu configuration. The COM Based Transformer can be configured according to the information from the COM module Ecu configuration.
Dependencies:	–
Use Case:	Inter-ECU Sender/Receiver communication based on fixed communication matrix Inter-ECU Client/Server communication
Supporting Material:	[5]

⌋*(RS_BRF_01316, RS_BRF_01544, RS_BRF_01560, RS_BRF_01592)*

5 References

References

[1] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[2] Glossary
AUTOSAR_TR_Glossary

[3] Requirements on AUTOSAR Features
AUTOSAR_RS_Features

[4] System Template
AUTOSAR_TPS_SystemTemplate

[5] Specification of Communication
AUTOSAR_SWS_COM