

<b>Document Title</b>	Specification of a Transport Layer for SAE J1939
<b>Document Owner</b>	AUTOSAR
<b>Document Responsibility</b>	AUTOSAR
<b>Document Identification No</b>	425

<b>Document Status</b>	published
<b>Part of AUTOSAR Standard</b>	Classic Platform
<b>Part of Standard Release</b>	R25-11

Document Change History			
Date	Release	Changed by	Description
2025-11-27	R25-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Support for back-to-back routing</li> <li>• Support for non-default priorities and BAM timings</li> <li>• Removed <code>J1939Tp_ChangeParameter()</code></li> <li>• Improved uptracing to RS BSW General</li> </ul>
2024-11-27	R24-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Changed lower layer to LSduR</li> </ul>
2023-11-23	R23-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Changed document name to include "CP"</li> <li>• Added references to <code>ComMChannels</code> to configuration</li> <li>• Added information about automatic handle IDs to configuration</li> </ul>
2022-11-24	R22-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Fixed not applicable SWS items</li> </ul>
2021-11-25	R21-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• No content changes</li> </ul>
2020-11-30	R20-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Improved structure of error sections</li> <li>• Replaced error descriptions with generated tables</li> </ul>





2019-11-28	R19-11	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• No content changes</li> <li>• Changed Document Status from Final to published</li> </ul>
2018-10-31	4.4.0	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Changed header file structure</li> <li>• Fixed reference to protocol type</li> <li>• Harmonized <a href="#">J1939TP_E_UNINIT</a></li> <li>• Clarifications of timeouts</li> </ul>
2017-12-08	4.3.1	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Clarified reception with identical channels</li> <li>• Introduction of runtime errors</li> </ul>
2016-11-30	4.3.0	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Improved handling of meta data</li> <li>• Reliable TxConfirmation replaces timeout</li> <li>• Clarification on concurrent connections</li> <li>• Support of link time configuration variant</li> </ul>
2015-07-31	4.2.2	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Fixed retry behavior</li> <li>• Clarified effect of meta data on SA/DA/Protocol</li> <li>• Introduction of further error classes</li> </ul>
2014-10-31	4.2.1	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Removed obsolete configuration elements</li> <li>• Harmonized with SWS BSW General</li> </ul>
2014-03-31	4.1.3	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• Allow for parallel reception via TP (BAM/CMDT) and directly</li> <li>• Clarified ambiguities regarding CTS parameters</li> </ul>



△

2013-10-31	4.1.2	AUTOSAR Release Management	<ul style="list-style-type: none"> <li>• <code>NotifResultType</code> replaced by <code>StdReturnType</code></li> <li>• Additional development errors for protocol problems</li> <li>• Connection parameters moved to channel configuration</li> <li>• Editorial changes</li> <li>• Removed chapter(s) on change documentation</li> </ul>
2013-03-15	4.1.1	AUTOSAR Administration	<ul style="list-style-type: none"> <li>• Support for dynamic CAN IDs via meta data</li> <li>• Support for variable block size and retry</li> <li>• TP API improvements and fixes</li> <li>• Adapted to new BSW General</li> </ul>
2011-12-22	4.0.3	AUTOSAR Administration	<ul style="list-style-type: none"> <li>• API changes: <code>CancelTransmit</code>, <code>CancelReceive</code>, and <code>ChangeParameter</code> were changed to synchronous behavior</li> </ul>
2011-04-15	4.0.2	AUTOSAR Administration	<ul style="list-style-type: none"> <li>• Fixed service API IDs</li> <li>• Removed duplicate requirement J1939TP0099</li> <li>• Clarified requirements [<a href="#">SWS_J1939Tp_00125</a>] and [<a href="#">SWS_J1939Tp_00189</a>]</li> </ul>
2009-12-18	4.0.1	AUTOSAR Administration	<ul style="list-style-type: none"> <li>• Initial release</li> </ul>

## 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	Introduction and Functional Overview	8
2	Glossary, Acronyms, and Abbreviations	10
3	Related Documentation	12
3.1	Input Documents & Related Standards and Norms	12
3.2	Related Specification	13
4	Constraints and Assumptions	14
4.1	Limitations	14
4.2	Applicability to Automotive Domains	14
5	Dependencies to Other Modules	15
5.1	File Structure	15
5.1.1	Code File Structure	16
5.1.2	Header File Structure	16
6	Requirements Tracing	17
7	Functional Specification	20
7.1	Basic Principles of SAE J1939	20
7.2	Basic Functionality of J1939Tp	21
7.3	Initialization and Shutdown	22
7.4	Communication with the Lower Layer	23
7.4.1	Transmission Confirmation	24
7.4.2	Reception Indication	24
7.5	Internal Behavior	24
7.5.1	Session Handling	24
7.5.1.1	Close Transmission	24
7.5.1.2	Abort Transmission	24
7.5.1.3	Close Reception	25
7.5.1.4	Abort Reception	25
7.5.1.5	Abort CMDT Connection	25
7.5.2	N-SDU Reception	25
7.5.3	N-SDU Transmission	27
7.5.4	Data Flow on the CAN Bus	28
7.5.4.1	Data Flow using Direct Transmission	28
7.5.4.2	Data Flow using CMDT	28
7.5.4.3	Data Flow using BAM	32
7.5.5	N-SDU Buffer Management	33
7.5.6	Relationship between N-SDU and N-PDU in J1939Tp	34
7.5.7	Concurrent Connections	35
7.5.8	N-PDU Padding	36

7.5.9	Handling of Unexpected N-PDU Arrivals	36
7.6	Error Classification	36
7.6.1	Development Errors	37
7.6.2	Runtime Errors	37
7.6.3	Production Errors	38
7.6.4	Extended Production Errors	38
8	API Specification	39
8.1	API Parameter Checking	39
8.2	Imported Types	39
8.3	Type Definitions	39
8.4	Function Definitions	40
8.4.1	J1939Tp_Init	40
8.4.2	J1939Tp_Shutdown	42
8.4.3	J1939Tp_GetVersionInfo	42
8.4.4	J1939Tp_Transmit	43
8.4.5	J1939Tp_CancelTransmit	44
8.4.6	J1939Tp_CancelReceive	44
8.5	Callback Notifications	45
8.5.1	J1939Tp_RxIndication	45
8.5.2	J1939Tp_TxConfirmation	46
8.6	Scheduled Functions	47
8.6.1	J1939Tp_MainFunction	47
8.7	Expected Interfaces	47
8.7.1	Mandatory Interfaces	47
8.7.2	Optional Interfaces	48
9	Sequence Diagrams	49
9.1	Reception of Direct PG	49
9.2	Reception via BAM	49
9.3	Reception via CMDT	50
9.4	Transmission of Direct PG	50
9.5	Transmission via BAM	51
9.6	Transmission via CMDT	51
9.7	Handling of Retry during CMDT Transmission	52
10	Configuration Specification	54
10.1	Containers and Configuration Parameters	54
10.1.1	J1939Tp	60
10.1.2	J1939TpGeneral	61
10.1.3	J1939TpConfiguration	63
10.1.4	J1939TpRxChannel	64
10.1.5	J1939TpRxCmNPdu	69
10.1.6	J1939TpRxDtNPdu	70
10.1.7	J1939TpRxPg	71

10.1.8 J1939TpRxDirectNPdu . . . . .	73
10.1.9 J1939TpRxNSdu . . . . .	74
10.1.10 J1939TpTxFcNPdu . . . . .	75
10.1.11 J1939TpTxChannel . . . . .	76
10.1.12 J1939TpRxFcNPdu . . . . .	82
10.1.13 J1939TpTxCmNPdu . . . . .	84
10.1.14 J1939TpTxDtNPdu . . . . .	85
10.1.15 J1939TpTxPg . . . . .	86
10.1.16 J1939TpTxDirectNPdu . . . . .	88
10.1.17 J1939TpTxNSdu . . . . .	89
A Not Applicable Requirements . . . . .	91
B Change History of AUTOSAR Traceable Items . . . . .	92
B.1 Traceable Item History of this Document According to AUTOSAR Re- lease R25-11 . . . . .	92
B.1.1 Added Specification Items in R25-11 . . . . .	92
B.1.2 Changed Specification Items in R25-11 . . . . .	92
B.1.3 Deleted Specification Items in R25-11 . . . . .	92
B.2 Traceable Item History of this Document According to AUTOSAR Re- lease R24-11 . . . . .	92
B.2.1 Added Specification Items in R24-11 . . . . .	92
B.2.2 Changed Specification Items in R24-11 . . . . .	92
B.2.3 Deleted Specification Items in R24-11 . . . . .	92

# 1 Introduction and Functional Overview

This specification describes the functionality, the API, and the configuration of the AUTOSAR Basic Software module J1939Tp, which implements a transport layer for AUTOSAR which is compatible to SAE J1939. The terms J1939Tp and SAE J1939 Transport Layer module are used synonymously in this document.

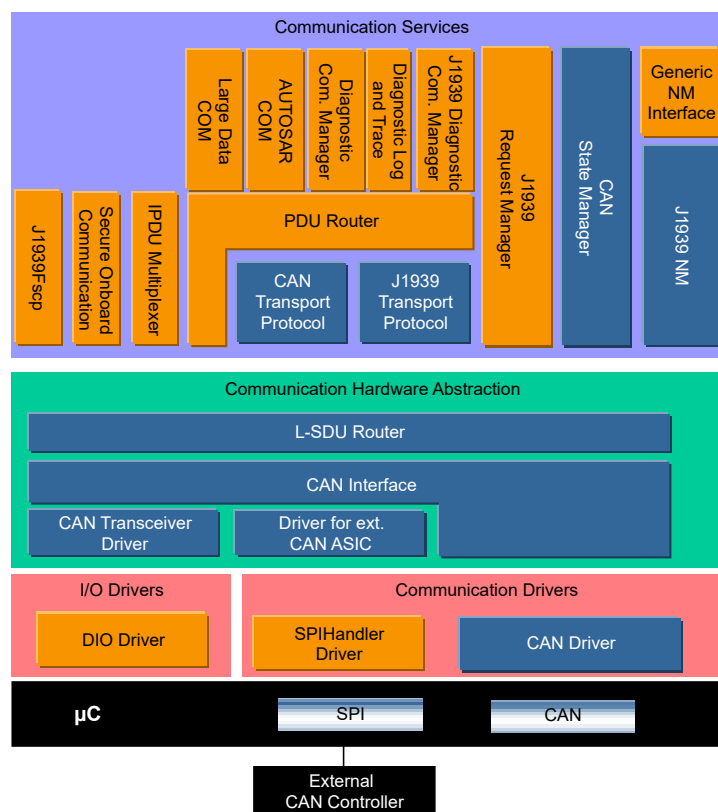
SAE J1939 has a broad acceptance in the truck domain, and consists of several documents describing the layers of the communication protocol from the physical layer to diagnostics and the application layer. SAE J1939-21 describes the data link and transport layer, which includes two transport protocol variants:

BAM (Broadcast Announce Message) for broadcast messages, and

CMDT (Connection Mode Data Transfer) for point-to-point connections.

This specification defines how the transport protocol of SAE J1939-21 can be implemented in the AUTOSAR architecture. It only describes those parts of the implementation that are relevant to the AUTOSAR architecture. Protocol specific details like exact timings are not part of this specification. Therefore, to be able to implement the SAE J1939 Transport Layer module, the reader of this specification needs to read also the original SAE J1939-21 specification document.

The module J1939Tp interfaces to the PDU Router and the CAN Interface (via the L-SDU Router), as shown in the following figure:



**Figure 1.1: AUTOSAR Communication Stack**



The purpose of [J1939Tp](#) is to segment and reassemble J1939 [PGNs](#) (N-SDUs) that are longer than 8 bytes. The segmented messages are sent and received via the [L-SDU Router](#).

The [PDU Router](#) ([PduR](#)) deploys AUTOSAR COM and DCM I-PDUs onto different communication protocols (e.g. CAN or J1939, segmented via transport layer or directly). At run-time, the [PduR](#) decides where to route I-PDUs based on the I-PDU (L-SDU or N-SDU) identifier. In the same way, the [LSduR](#) uses the L-SDU (I-PDU or N-PDU) identifier to decide whether a received message must be processed by one of the available transport layer modules or may be forwarded directly to the [PduR](#).

[J1939Tp](#) supports fixed and variable size N-SDUs (I-PDUs) with more than 8 bytes. I-PDUs that do not exceed 8 bytes are exchanged directly between [PduR](#) and [CanIf](#) via [LSduR](#).

Fixed size N-SDUs are always segmented by the [J1939Tp](#), while variable size N-SDUs are only segmented when they exceed 8 bytes. [J1939Tp](#) forwards variable size N-SDUs with an actual size of 8bytes or less and a configured maximum size of more than 8 bytes directly to the [LSduR](#). The [transport protocol](#) variant ([BAM](#) or [CMDT](#)) is chosen based on received N-PDUs when a large N-SDU is received, and on the configuration and the actual DA when a large N-SDU is transmitted.

[J1939Tp](#) supports handling of N-PDUs and N-SDUs with variable [SA](#), [DA](#), and Priority. In this case, the N-PDUs and N-SDUs will contain parts of the CAN ID in the [meta data](#).

In summary, [J1939Tp](#) provides the following functionality:

- Segmentation and direct transmission of data in transmit direction
- Reassembling and direct reception of data in receive direction
- Control of data flow
- Timeout supervision
- Detection of errors during segmentation or reassembly

## 2 Glossary, Acronyms, and Abbreviations

The glossary below includes terms and acronyms and abbreviations relevant to the [SAE J1939 Transport Layer](#) module that are not included in the [1, AUTOSAR Glossary].

Glossary Term	Explanation
Meta Data	Meta data transferred alongside a <a href="#">PDU</a> , consisting of a set of <a href="#">meta data items</a>
Meta Data Item	A single item of <a href="#">meta data</a> of defined type and size
Parameter	A parameter is a signal of the <a href="#">SAE J1939</a> application layer. The Parameter definition includes defined scale, limits, offset, and physical unit. Parameters are uniquely identified by the <a href="#">SPN</a> .
Parameter Group	A Parameter Group is a message of the <a href="#">SAE J1939</a> application layer. Each Parameter Group contains several <a href="#">Parameters</a> (signals), and is uniquely identified by the <a href="#">PGN</a> .
Parameter Group Number	Unique identifier (18 bits: <a href="#">EDP</a> , <a href="#">DP</a> , <a href="#">PDUF</a> , <a href="#">PDUS</a> ) of an <a href="#">SAE J1939 Parameter Group</a> that is contained in the payload of many <a href="#">SAE J1939</a> protocol messages and in the 29bit CAN identifier of <a href="#">SAE J1939 PGs</a> .
SAE J1939	Serial control and communications standard for heavy duty vehicle networks created by the <a href="#">SAE</a> , see [2, SAE J1939]
SAE J1939-21	Data link layer for CAN 2.0 created by the <a href="#">SAE</a> , see [3, SAE J1939-21]
SAE J1939-73	Diagnostics application layer created by the <a href="#">SAE</a> , see [4, SAE J1939-73]
SAE J1939-81	Network management layer created by the <a href="#">SAE</a> , see [5, SAE J1939-81]
SAE J1939 Address Claiming	The <a href="#">SAE J1939</a> address claiming procedure is the core feature of the network management of <a href="#">SAE J1939</a> defined in the standard document <a href="#">SAE J1939-81</a> . Address claiming assigns a temporary 8-bit identifier to each <a href="#">CAs</a> connected to an <a href="#">SAE J1939</a> network, based on the <a href="#">NAME</a> of the <a href="#">CAs</a> . Within each network, the 8-bit identifier is required to be unique. The 8-bit identifier is used as <a href="#">SA</a> and <a href="#">DA</a> of <a href="#">Parameter Groups</a> (messages) transferred via the <a href="#">SAE J1939</a> network. The address claiming procedure is based on the exchange of <a href="#">AddressClaimed</a> messages.
SAE J1939 Diagnostics	The <a href="#">SAE J1939</a> diagnostic layer is defined in the standard document <a href="#">SAE J1939-73</a> . The <a href="#">SAE J1939</a> diagnostics is functionally similar to the UDS diagnostics, and has recently been extended to support OBD for emission related values.
SAE J1939 Transport Protocol	The <a href="#">SAE J1939</a> transport protocol is used for the segmented transmission of <a href="#">PGs</a> with more than 8 bytes of data. The transport protocol is defined in the network layer standard document <a href="#">SAE J1939-21</a> .

**Table 2.1: Glossary Terms**

Acronym / Abbreviation	Description
AC	<a href="#">SAE J1939 AddressClaimed PG</a> ( <a href="#">PGN</a> = 0x0EE00), which contains the <a href="#">NAME</a> and is used for the <a href="#">SAE J1939 address claiming</a> , CannotClaimAddress when <a href="#">SA</a> = 0xFE
BAM	Broadcast Announce Message, broadcast variant of <a href="#">SAE J1939 transport protocol</a>
BSW	Basic Software module
CA	Controller Application, role of an ECU tied to one address
Can	CAN Driver, abstracting from CAN controller hardware, see [6, SWS CAN Driver]
CanIf	CAN Interface, abstracting from CAN drivers and transceivers, see [7, SWS CAN Interface]
CanTp	CAN Transport Layer, see [8, SWS CAN Transport Layer]
CMDT	Connection Mode Data Transfer, a.k.a. <a href="#">RTS/CTS</a> , peer-to-peer variant of <a href="#">SAE J1939 transport protocol</a>
DA	Destination Address, the address of the receiving <a href="#">CA</a> of a <a href="#">PG</a> . Part of the 29 bit CAN identifier of <a href="#">SAE J1939 PGs</a> .
DET	Default Error Tracer, supports development and runtime error reporting, see [9, SWS Default Error Tracer]





Acronym / Abbreviation	Description
DP	Data Page, the most significant bit (MSB) of the 18 bit <a href="#">PGN</a>
EcuM	ECU State Manager, mainly responsible for startup and shutdown of the ECU, see [10, SWS ECU State Manager]
EDP	Extended Data Page, the second bit (after MSB) of the 18 bit <a href="#">PGN</a>
FrTp	FlexRay ISO Transport Layer, see [11, SWS FlexRay ISO Transport Layer]
J1939Dcm	<a href="#">SAE J1939</a> Diagnostic Communication Manager, implementing the <a href="#">SAE J1939 Diagnostics</a> , see [12, SWS SAE J1939 Diagnostic Communication Manager]
J1939Tp	<a href="#">SAE J1939</a> Transport Layer, this <a href="#">BSW</a> module, implementing the <a href="#">SAE J1939 Transport Protocol</a>
LSduR	L-SDU Router, establishes communication between bus interface and other <a href="#">BSW</a> modules, see [13, SWS L-SDU Router]
NAME	The 64 bit NAME of a <a href="#">CA</a> used in the <a href="#">AddressClaimed PG</a>
PDU	Protocol Data Unit, a message transferred between the layers of the AUTOSAR stack, also known as I-PDU
PDU1	<a href="#">SAE J1939</a> PDU Type 1, this kind of <a href="#">PG</a> can be sent to a specific <a href="#">destination address</a>
PDU2	<a href="#">SAE J1939</a> PDU Type 2, this kind of <a href="#">PG</a> can only be sent as broadcast
PDUF	PDU Format, the middle byte of the 18 bit <a href="#">PGN</a> which identifies the <a href="#">PG</a> and determines the layout ( <a href="#">PDU1/PDU2</a> ) of the <a href="#">PGN</a>
PduR	PDU Router, establishes communication between <a href="#">BSW</a> modules, see [14, SWS PDU Router]
PDUS	PDU Specific, the lower byte of the 18 bit <a href="#">PGN</a> which further identifies <a href="#">PDU2 PG</a> which do cannot have a <a href="#">destination address</a>
PG	<a href="#">Parameter Group</a> , corresponds to an N-SDU of <a href="#">J1939Tp</a>
PGN	<a href="#">Parameter Group Number</a>
RTE	AUTOSAR Runtime Environment, handles application <a href="#">SW-Cs</a> and interfaces between <a href="#">basic software</a> and the application, see [15, SWS RTE]
SA	Source Address, the address of the transmitting <a href="#">CA</a> of a <a href="#">PG</a> . Part of the 29 bit CAN identifier of <a href="#">SAE J1939 PGs</a> .
SAE	Society of Automotive Engineers (in charge of <a href="#">SAE J1939</a> specification)
SchM	Basic Software Schedule Manager, part of the <a href="#">RTE</a>
SNA	Signal Not Available, all bits set to 1 in <a href="#">SAE J1939 PGs/Parameters</a>
SPN	Suspect Parameter Number, unique identifier of an <a href="#">SAE J1939 Parameter</a>
SW-C	AUTOSAR Software Component (of the Application)
TP.CM	Connection Management message ( <a href="#">PGN</a> = 0x0EC00) used by <a href="#">SAE J1939 transport protocol</a> , corresponds to an N-PDU of <a href="#">J1939Tp</a>
TP.CM_BAM	<a href="#">Broadcast Announce Message</a> , variant of <a href="#">TP.CM</a> that initiates a <a href="#">BAM</a> transmission
TP.CM_CTS	<a href="#">Connection Mode Clear To Send</a> , variant of <a href="#">TP.CM</a> that is used for handshake during <a href="#">CMDT</a> transmission
TP.CM_EndOfMsgAck	End Of Message Acknowledge, variant of <a href="#">TP.CM</a> that acknowledges correct reception of a <a href="#">CMDT</a> transmission
TP.CM_RTS	<a href="#">Connection Mode Request To Send</a> , variant of <a href="#">TP.CM</a> that initiates a <a href="#">CMDT</a> transmission
TP.Conn_Abort	Connection Abort, variant of <a href="#">TP.CM</a> that terminates a <a href="#">CMDT</a> transmission
TP.DT	Data Transfer message ( <a href="#">PGN</a> = 0x0EB00) used by <a href="#">SAE J1939 transport protocol</a> , corresponds to an N-PDU of <a href="#">J1939Tp</a>

**Table 2.2: Acronyms and Abbreviations**

## 3 Related Documentation

### 3.1 Input Documents & Related Standards and Norms

- [1] Glossary  
AUTOSAR\_FO\_TR\_Glossary
- [2] SAE J1939 – Serial Control and Communications Heavy Duty Vehicle Network
- [3] SAE J1939-21 Data Link Layer
- [4] SAE J1939-73 Application Layer – Diagnostics
- [5] SAE J1939-81 Network Management
- [6] Specification of CAN Driver  
AUTOSAR\_CP\_SWS\_CANDriver
- [7] Specification of CAN Interface  
AUTOSAR\_CP\_SWS\_CANInterface
- [8] Specification of CAN Transport Layer  
AUTOSAR\_CP\_SWS\_CANTransportLayer
- [9] Specification of Default Error Tracer  
AUTOSAR\_CP\_SWS\_DefaultErrorTracer
- [10] Specification of ECU State Manager  
AUTOSAR\_CP\_SWS\_ECUSTateManager
- [11] Specification of FlexRay ISO Transport Layer  
AUTOSAR\_CP\_SWS\_FlexRayISOTransportLayer
- [12] Specification of a Diagnostic Communication Manager for SAE J1939  
AUTOSAR\_CP\_SWS\_SAEJ1939DiagnosticCommunicationManager
- [13] Specification of Linklayer Sdu Routing Module  
AUTOSAR\_CP\_SWS\_LSduRouter
- [14] Specification of PDU Router  
AUTOSAR\_CP\_SWS\_PDURouter
- [15] Specification of RTE Software  
AUTOSAR\_CP\_SWS\_RTE
- [16] General Specification of Basic Software Modules  
AUTOSAR\_CP\_SWS\_BSWGeneral
- [17] Layered Software Architecture  
AUTOSAR\_CP\_EXP\_LayeredSoftwareArchitecture
- [18] Requirements on BSW Modules for SAE J1939  
AUTOSAR\_CP\_RS\_SAEJ1939

- [19] General Requirements on Basic Software Modules  
AUTOSAR\_CP\_RS\_BSWGeneral
- [20] System Template  
AUTOSAR\_CP\_TPS\_SystemTemplate
- [21] Specification of ECU Configuration  
AUTOSAR\_CP\_TPS\_ECUConfiguration

## 3.2 Related Specification

AUTOSAR provides a General Specification on [Basic Software](#) modules [[16](#), SWS BSW General], which is also valid for [SAE J1939 Transport Layer](#).

Thus, the specification [[16](#), SWS BSW General] shall be considered as additional and required specification for [SAE J1939 Transport Layer](#).

## 4 Constraints and Assumptions

### 4.1 Limitations

The AUTOSAR architecture contains several communication system specific transport layers ([J1939Tp](#), [CanTp](#), [FrTp](#), etc.). All of these modules need to have identical APIs, with the exception of API functions for which the [PduR](#) has separate configuration abilities.

The [J1939Tp](#) module does not implement the `TriggerTransmit` API, because it is only needed for time triggered bus architectures.

### 4.2 Applicability to Automotive Domains

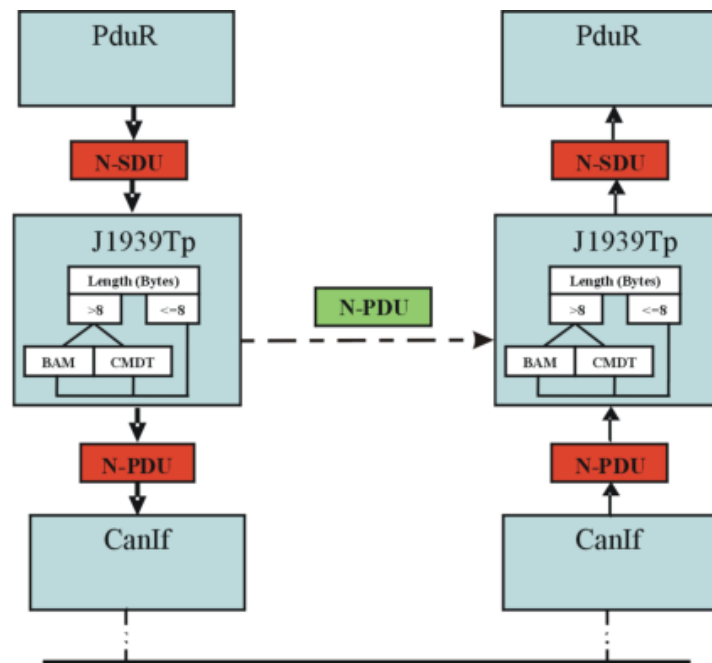
The [SAE J1939 Transport Layer](#) supports the implementation of ECUs that are designed to operate in a J1939 network.

## 5 Dependencies to Other Modules

The [17, EXP Layered Software Architecture] shows an overview of the neighboring modules of the [SAE J1939 Transport Layer](#).

Besides the standard modules [Default Error Tracer \(DET\)](#), [ECU State Manager \(EcuM\)](#), and [Basic Software Scheduler \(SchM\)](#), which have interfaces to all BSW modules, [J1939Tp](#) only interacts with the [PDU Router \(PduR\)](#) and the [L-SDU Router \(LSduR\)](#), and indirectly with the [CAN Interface \(CanIf\)](#). The interfaces of [J1939Tp](#) are similar to the interfaces of [CAN Transport Layer \(CanTp\)](#).

The figure below shows the interactions between [J1939Tp](#), [PduR](#), and [CanIf](#), omitting [LSduR](#), [CAN Driver](#), and potential upper layers, which are not relevant to understand these interactions.



**Figure 5.1: J1939 Transport Layer interactions**

The [J1939Tp](#)'s upper interface offers the [PduR](#) module access to transmitted and received N-SDUs corresponding to J1939 [PGs](#) with a maximum length of more than 8 bytes of data.

J1939 [PGs](#) with a maximum length of 8 bytes or less are exchanged directly between [PduR](#) and [CanIf](#) via [LSduR](#).

### 5.1 File Structure

AUTOSAR specifies that an ECU can be created from modules provided as object code, source code (generated or static), or both.

The decision to provide a module as object code or source code is based on a compromise between IP protection, test coverage, code efficiency and configurability at system generation time. Depending on the configurability requirements of the OEM, suppliers may deliver the [J1939Tp](#) module as object code or source code.

The file hierarchy defined in this section allows the separation of platform, compiler, and implementation specific definitions and declarations from general definitions, as well as the separation of source code and configuration.

### 5.1.1 Code File Structure

For details, refer to [[16](#), SWS BSW General] 5.1.6 “*Code file structure*”.

**[SWS\_J1939Tp\_00007]** [Internally used data types and functions shall be defined locally in the implementation source files.]

This prevents visibility of these symbols outside of the [J1939Tp](#) module.

### 5.1.2 Header File Structure

For details, refer to [[16](#), SWS BSW General] 5.1.7 “*Header file structure*”.



## 6 Requirements Tracing

The following tables reference the requirements specified in [18, SRS SAE J1939] (Requirements on BSW Modules for SAE J1939) and [19, SRS BSW General] and links to the fulfillment of these.

Requirement	Description	Satisfied by
[SRS_BSW_00101]	The Basic Software Module shall be able to initialize variables and hardware in a separate initialization function	[SWS_J1939Tp_00087]
[SRS_BSW_00167]	All AUTOSAR Basic Software Modules shall provide configuration rules and constraints to enable plausibility checks	[SWS_J1939Tp_00084]
[SRS_BSW_00171]	Optional functionality of a Basic-SW component that is not required in the ECU shall be configurable at pre-compile-time	[SWS_J1939Tp_00125]
[SRS_BSW_00323]	All AUTOSAR Basic Software Modules shall check passed API parameters for validity	[SWS_J1939Tp_00188]
[SRS_BSW_00327]	Error values naming convention	[SWS_J1939Tp_00115]
[SRS_BSW_00333]	For each callback function it shall be specified if it is called from interrupt context or not	[SWS_J1939Tp_00110] [SWS_J1939Tp_00114]
[SRS_BSW_00335]	Status values naming convention	[SWS_J1939Tp_00019]
[SRS_BSW_00336]	Basic SW module shall be able to shutdown	[SWS_J1939Tp_00093]
[SRS_BSW_00337]	Classification of development errors	[SWS_J1939Tp_00115]
[SRS_BSW_00357]	For success/failure of an API call a standard return type shall be defined	[SWS_J1939Tp_00096]
[SRS_BSW_00358]	The return type of init() functions implemented by AUTOSAR Basic Software Modules shall be void	[SWS_J1939Tp_00087]
[SRS_BSW_00359]	Callback Function Return Types for AUTOSAR BSW	[SWS_J1939Tp_00108] [SWS_J1939Tp_00112]
[SRS_BSW_00360]	AUTOSAR Basic Software Modules callback functions are allowed to have parameters	[SWS_J1939Tp_00108] [SWS_J1939Tp_00112]
[SRS_BSW_00373]	The main processing function of each AUTOSAR Basic Software Module shall be named according the defined convention	[SWS_J1939Tp_00104]
[SRS_BSW_00385]	List possible error notifications	[SWS_J1939Tp_00115]
[SRS_BSW_00386]	The BSW shall specify the configuration and conditions for detecting an error	[SWS_J1939Tp_00026] [SWS_J1939Tp_00071] [SWS_J1939Tp_00188]
[SRS_BSW_00400]	Parameter shall be selected from multiple sets of parameters after code has been loaded and started	[SWS_J1939Tp_00187]
[SRS_BSW_00405]	BSW Modules shall support multiple configuration sets	[SWS_J1939Tp_00187]
[SRS_BSW_00406]	API handling in uninitialized state	[SWS_J1939Tp_00023]





Requirement	Description	Satisfied by
[SRS_BSW_00407]	Each BSW module shall provide a function to read out the version information of a dedicated module implementation	[SWS_J1939Tp_00089]
[SRS_BSW_00414]	Init functions shall have a pointer to a configuration structure as single parameter	[SWS_J1939Tp_00187]
[SRS_BSW_00438]	Configuration data shall be defined in a structure	[SWS_J1939Tp_00187]
[SRS_BSW_00441]	Naming convention for type, macro and function	[SWS_J1939Tp_00019] [SWS_J1939Tp_00115]
[SRS_BSW_00450]	A Main function of a un-initialized module shall return immediately	[SWS_J1939Tp_00023]
[SRS_BSW_00478]	Timing limits of main functions	[SWS_J1939Tp_00106]
[SRS_J1939_00001]	The J1939 Transport Layer module shall be configurable to support only transport protocol variant BAM	[SWS_J1939Tp_00125]
[SRS_J1939_00006]	The properties of a J1939Tp N-SDU shall be statically configured	[SWS_J1939Tp_00125]
[SRS_J1939_00010]	The J1939 Transport Layer module shall implement an interface for initialization	[SWS_J1939Tp_00020] [SWS_J1939Tp_00024]
[SRS_J1939_00011]	The J1939 Transport Layer services shall not be operational before initializing the module	[SWS_J1939Tp_00019] [SWS_J1939Tp_00020] [SWS_J1939Tp_00023] [SWS_J1939Tp_00076]
[SRS_J1939_00018]	The AUTOSAR J1939 Transport Layer module shall support concurrent connections	[SWS_J1939Tp_00231] [SWS_J1939Tp_00233]
[SRS_J1939_00019]	The J1939 Transport Layer module shall support the transport protocol variant BAM	[SWS_J1939Tp_00155] [SWS_J1939Tp_00236]
[SRS_J1939_00020]	The AUTOSAR J1939 Transport Layer module shall support the transport protocol variant CMDT	[SWS_J1939Tp_00155] [SWS_J1939Tp_00236]
[SRS_J1939_00022]	The timeout values of the J1939 transport protocol variants shall be supervised	[SWS_J1939Tp_00018] [SWS_J1939Tp_00121] [SWS_J1939Tp_00123]
[SRS_J1939_00023]	The J1939 Transport Layer module shall handle unexpected PDUs according to the SAE J1939 specification	[SWS_J1939Tp_00018] [SWS_J1939Tp_00064]
[SRS_J1939_00024]	Unused Bytes in N-PDUs shall be padded	[SWS_J1939Tp_00068]
[SRS_J1939_00025]	The J1939 Transport Layer module shall be able to manage connections via BAM and CMDT in parallel	[SWS_J1939Tp_00062]
[SRS_J1939_00040]	The AUTOSAR J1939 Transport Layer module shall be based on SAE J1939 specifications	[SWS_J1939Tp_00018]
[SRS_J1939_00042]	The J1939 Transport Layer interface shall be independent of its internal configuration	[SWS_J1939Tp_00157]
[SRS_J1939_00044]	The J1939 Transport Layer module shall support generic N-SDUs	[SWS_J1939Tp_00233]





Requirement	Description	Satisfied by
[SRS_J1939_00049]	J1939 Modules shall support Meta Data	[SWS_J1939Tp_00198] [SWS_J1939Tp_00236]
[SRS_J1939_00067]	The J1939 Transport Layer module shall support buffering transmission requests for N-SDUs that share resources with an ongoing transmission	[SWS_J1939Tp_00237] [SWS_J1939Tp_00238]

**Table 6.1: Requirements Tracing**

## 7 Functional Specification

This chapter describes the functionality of the [SAE J1939 Transport Layer](#). It explains the services provided to the upper and lower layers and the internal behavior of [J1939Tp](#).

### 7.1 Basic Principles of SAE J1939

The SAE J1939 standard defines a set of [Parameter Groups](#) (PGs), each containing signals with defined content and semantics. The following information is provided for each PG:

- Payload length type: maximal number of bytes, fixed or variable size.
- [Parameter Group Number](#) (PGN): 18 bit value containing the following fields:
  - 2 bit data page information ([EDP](#) and [DP](#))
  - 8 bit [PDU Format](#) (PDUF)
  - 8 bit [PDU Specific](#) (PDUS)

PGNs with [PDU Format](#) < 240 ([PDU1](#) format) identify point-to-point messages, while PGNs with [PDU Format](#) ≥ 240 ([PDU2](#) format) identify broadcast messages. The [PDU Specific](#) field is only relevant for broadcast messages ([PDU2](#) format); it is always zero for point-to-point messages ([PDU1](#) format).

J1939 uses 29-bit CAN identifiers to identify each message. The CAN identifier contains a 3-bit-priority, the [PGN](#), the [Source Address](#) (SA), and the [Destination Address](#) (DA, only for point-to-point messages).

Usually, an ECU has just one node address, which is used as [DA](#) in received messages and as [SA](#) in transmitted messages. However, a single ECU can also implement several different J1939 nodes at once, each with its own node address. In this case, the ECU accepts any of these node addresses as [DA](#) and sends with any of the defined node addresses as [SA](#).

The [SAE J1939 Transport Layer](#) uses two special point-to-point messages identified by PGNs of [PDU1](#) format to transport segmented messages, both with a fixed length of 8 bytes. These messages are called transport frames in the context of this document.

[TP.CM](#) is used for connection management. The first byte of the payload identifies its role, which may be one of the following:

[TP.CM\\_BAM](#) is used to initiate a [BAM](#) transfer.

[TP.CM\\_RTS](#) is transmitted to initiate a [CMDT](#) transfer.

[TP.CM\\_CTS](#) is used for flow control during a [CMDT](#) transfer.

`TP.CM_EndOfMsgAck` indicates the end of a `CMDT` transfer.

`TP.Conn_Abort` indicates an error and terminates the `CMDT` transfer.

`TP.DT` contains a sequence number in the first byte and 7 bytes of data, used for both `BAM` and `CMDT` transfers.

A single `TP.CM` or `TP.DT` frame, identified by a certain CAN Identifier, is used for different `PGs`. The `PGN` of the transported `PG` is contained in the payload of the `TP.CM` frames as specified in SAE J1939-21.

The `destination address` (`DA`) of `CMDT` related transport frames contains a valid node address and thus allows a point-to-point connection, while the `DA` of `BAM` related transport frames is always set to `0xFF` to create a broadcast connection.

#### [SWS\_J1939Tp\_00018]

*Upstream requirements:* `SRS_J1939_00022`, `SRS_J1939_00023`, `SRS_J1939_00040`

[The `J1939Tp` module shall follow the recommendations of SAE J1939-21 if they are not explicitly excluded in this document.]

## 7.2 Basic Functionality of J1939Tp

This section describes aspects of the functionality of `J1939Tp` that are not related to neighboring modules.

#### [SWS\_J1939Tp\_00071]

*Upstream requirements:* `SRS_BSW_00386`

[On errors and exceptions, the `J1939Tp` module shall not modify its current module state but shall simply report the error event.]

See [Figure 7.1](#) for the `J1939Tp` module states.

To assure a unique `PDU Router` handling of all J1939 `PGs` which is independent of the Payload Length type (variable or fixed), the `J1939Tp` is used for the transmission of all `Parameter Groups` that are longer than 8 Bytes, independent of their length being fixed or variable.

This means that `PGs` with variable length that have a configured maximum size of more than 8 bytes but do not exceed 8 bytes at run-time shall be transported using `J1939Tp` even though no segmentation is necessary and a direct transmission from `PDU Router` to `CAN Interface` via `L-SDU Router` would be possible.

#### [SWS\_J1939Tp\_00155]

*Upstream requirements:* `SRS_J1939_00019`, `SRS_J1939_00020`

[The SAE J1939 Transport Layer shall implement the following two `J1939 transport protocol` variants defined in SAE J1939-21:

- [BAM](#) for broadcast transmission
- [CMDT](#) for point-to-point transmission

]

#### [SWS\_J1939Tp\_00125]

*Upstream requirements:* [SRS\\_BSW\\_00171](#), [SRS\\_J1939\\_00001](#), [SRS\\_J1939\\_00006](#)

[The [SAE J1939 Transport Layer](#) shall be configurable to either use both [BAM](#) and [CMDT transport protocol](#) variants, or only [BAM](#), or only [CMDT](#).]

#### [SWS\_J1939Tp\_00198]

*Upstream requirements:* [SRS\\_J1939\\_00049](#)

[The [SAE J1939 Transport Layer](#) shall use [meta data items](#) of type [SOURCE\\_ADDRESS\\_16](#), [TARGET\\_ADDRESS\\_16](#), and [PRIORITY\\_8](#) of all N-SDUs, and [meta data items](#) of type [CAN\\_ID\\_32](#) for all N-PDUs.]

#### [SWS\_J1939Tp\_00236] Configurable N-PDU Priorities

*Upstream requirements:* [SRS\\_J1939\\_00019](#), [SRS\\_J1939\\_00020](#), [SRS\\_J1939\\_00049](#)

[The [SAE J1939 Transport Layer](#) shall consider [J1939TpTxUseNonDefaultTpPriority](#), [J1939TpTxPriority](#), and [J1939TpTxDirectPriority](#) together with [meta data item](#) [PRIORITY\\_8](#) (see [[SWS\\_J1939Tp\\_00198](#)]) if these are configured.]

#### [SWS\_J1939Tp\_00231]

*Upstream requirements:* [SRS\\_J1939\\_00018](#)

[The [SAE J1939 Transport Layer](#) module shall support several connections for the same [PGN](#) in parallel as long as they can be received or transmitted independently.]

For the definition of connections see [Chapter 7.5.7](#).

#### [SWS\_J1939Tp\_00233]

*Upstream requirements:* [SRS\\_J1939\\_00018](#), [SRS\\_J1939\\_00044](#)

[If several receiving connections are configured with identical [PGNs](#) and matching channel parameters, the [J1939Tp](#) shall allocate an arbitrary free connection with matching parameters.]

Note: This scenario is typically only useful if all of these similar connections are configured to end up at the same upper layer module.

## 7.3 Initialization and Shutdown

The following figure summarizes all the requirements concerning initialization and shut down:

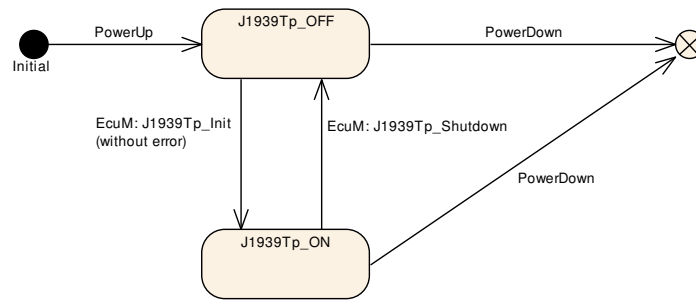


Figure 7.1: J1939 Transport Layer life cycle

#### [SWS\_J1939Tp\_00019]

Upstream requirements: [SRS\\_BSW\\_00335](#), [SRS\\_BSW\\_00441](#), [SRS\\_J1939\\_00011](#)

[The [J1939Tp](#) module shall have two global states, J1939TP\_ON and J1939TP\_OFF.]

#### [SWS\_J1939Tp\_00020]

Upstream requirements: [SRS\\_J1939\\_00010](#), [SRS\\_J1939\\_00011](#)

[The [J1939Tp](#) module shall be in the J1939TP\_OFF state after power up.]

In the state J1939TP\_OFF, the [J1939Tp](#) allows for an update of the post build configuration.

#### [SWS\_J1939Tp\_00023]

Upstream requirements: [SRS\\_BSW\\_00406](#), [SRS\\_BSW\\_00450](#), [SRS\\_J1939\\_00011](#)

[The [J1939Tp](#) module shall perform segmentation and reassembly tasks only when the [J1939Tp](#) is in the J1939TP\_ON state.]

#### [SWS\_J1939Tp\_00076]

Upstream requirements: [SRS\\_J1939\\_00011](#)

[The global state (see [[SWS\\_J1939Tp\\_00019](#)]) shall be used to check if the module has been initialized before calling an API.]

## 7.4 Communication with the Lower Layer

#### [SWS\_J1939Tp\_00157]

Upstream requirements: [SRS\\_J1939\\_00042](#)

[All service interfaces provided to the lower layer shall be independent of the internal communication configuration and implementation of the [J1939Tp](#).]

[SWS\_J1939Tp\_00041] [[J1939Tp](#) shall send the transport frames [TP.CM](#) and [TP.DT](#) and direct frames using the service function `LSduR_J1939TpTransmit()`.]

### 7.4.1 Transmission Confirmation

The transmission confirmation `J1939Tp_TxConfirmation()` is called by `CanIf` via `LSduR` to notify `J1939Tp` of successful or failed transmission of an N-PDU.

**[SWS\_J1939Tp\_00035]** [When `J1939Tp_TxConfirmation()` is called with `result E_NOT_OK`, the `J1939Tp` module shall abort the corresponding session.]

For transmitting sessions, the session abort is defined in **[SWS\_J1939Tp\_00032]**, while **[SWS\_J1939Tp\_00031]** defines the behavior in case of a receiving session. The `J1939Tp` shall not try to send an abort frame (as defined by **[SWS\_J1939Tp\_00097]**) in this case.

### 7.4.2 Reception Indication

The `J1939Tp` module shall provide a `J1939Tp_RxIndication()` API to allow the `CanIf` to notify via the `LSduR` that a new N-PDU has been received (see **[SWS\_J1939Tp\_00108]**).

`CanIf` shall perform the Reception Indication according to its configuration (i.e. in ISR context if configured so).

## 7.5 Internal Behavior

### 7.5.1 Session Handling

#### 7.5.1.1 Close Transmission

**[SWS\_J1939Tp\_00119]** [When the transport transmission session is successfully completed, the `J1939Tp` module shall call a notification service of the upper layer, `PduR_J1939TpTxConfirmation()`, with the `result E_OK`, to notify that the N-SDU transfer is successfully processed.]

#### 7.5.1.2 Abort Transmission

**[SWS\_J1939Tp\_00032]** [An Abort Transmission feature shall indicate the upper layer that the transmission of a given N-SDU has been aborted. It uses the callback function `PduR_J1939TpTxConfirmation()` with `E_NOT_OK`.]



### 7.5.1.3 Close Reception

**[SWS\_J1939Tp\_00118]** [A close connection feature shall indicate the upper layer that the transport reception session is completed. It uses the `PduR_J1939TpRxIndication()` with `E_OK` to indicate that the N-SDU reception is successfully processed.]

### 7.5.1.4 Abort Reception

**[SWS\_J1939Tp\_00031]** [An Abort Reception feature shall indicate the upper layer that the reception of a given N-SDU has been aborted. It uses the `PduR_J1939TpRxIndication()` with state `E_NOT_OK`.]

### 7.5.1.5 Abort CMDT Connection

**[SWS\_J1939Tp\_00097]** [An Abort CMDT Connection feature shall indicate to the other ECU participating in the concerned connection that the current CMDT session (transmission or reception) cannot be completed successfully. A `TP.Conn_Abort` abort message shall be transmitted to the other ECU via CAN as specified by SAE J1939-21.]

Sending a `TP.Conn_Abort` is necessary every time an error happens after `TP.CM_CTS` has been successfully sent or received, and is advisable already after transmission or reception of `TP.CM_RTS`. The connection abort reason of the `TP.Conn_Abort` shall be set according to SAE J1939-21.

## 7.5.2 N-SDU Reception

Reception of an N-SDU is always initiated by the reception of a `TP.CM` message. In case of CMDT, this is a `TP.CM_RTS`, in case of BAM a `TP.CM_BAM`.

**[SWS\_J1939Tp\_00043]** [Depending on the control byte of the initializing `TP.CM` frame (`TP.CM_BAM` or `TP.CM_RTS`), the `J1939Tp` module shall use the variant BAM or CMDT of the J1939 transport protocol to handle the data reception.]

**[SWS\_J1939Tp\_00038]** [When receiving an N-PDU containing a `TP.CM_BAM` or a `TP.CM_RTS` the `J1939Tp` module shall first notify the upper layer (PDU Router) before processing the frame reassembly. It uses the `PduR_J1939TpStartOfReception()` function with the following parameters:

- the Identifier of the corresponding N-SDU,
- depending on the configured meta data items: the SA, DA and Priority,

- the total Data Length (after reassembly) and
- a pointer to a location where the upper layer stores its currently available buffer size.

]

**[SWS\_J1939Tp\_00162]** [After the reception of an N-PDU containing a [TP.DT](#) frame, the function `PduR_J1939TpCopyRxData()` shall be called with the following parameters:

- the Identifier of the corresponding N-SDU,
- `info` with max. 7 bytes of data,
- a pointer to a location where the upper layer stores its currently available buffer size.

]

**[SWS\_J1939Tp\_00173]** [The [J1939Tp](#) module shall abort the reception silently if any of the following conditions occurs:

- The `PduR_J1939TpStartOfReception()` function returns `BUFREQ_E_NOT_OK` or `BUFREQ_E_OVFL` or
- the protocol chosen in [\[SWS\\_J1939Tp\\_00043\]](#) does not match the transport protocol configured for the transported `PGN` (see [J1939TpRxProtocolType](#)) or
- the `SA/DA` of N-PDUs with `meta data` do not match the configured `SA/DA` (see [J1939TpRxSa/J1939TpRxDa](#)).

In case of a [BAM](#) connection, no further activity is required.

In case of a [CMDT](#) connection, a [CMDT Connection Abort](#) shall be performed as described in [\[SWS\\_J1939Tp\\_00097\]](#), and the connection abort reason shall be set to 1 (Already in one or more connection managed sessions and cannot support another).]

**[SWS\_J1939Tp\_00040]** [The [J1939Tp](#) module shall abort the reception like indicated in [\[SWS\\_J1939Tp\\_00031\]](#) if any of the following conditions occurs:

- The value returned by `PduR_J1939TpStartOfReception()` via `bufferSizePtr` is smaller than the total data length of the N-SDU when received via the direct N-PDU,
- the `PduR_J1939TpCopyRxData()` function returns `BUFREQ_E_NOT_OK`, or
- the [J1939Tp\\_CancelReceive\(\)](#) function is called.

In case of a [BAM](#) connection, no further activity is required.

In case of a [CMDT](#) connection, a [CMDT Connection Abort](#) shall be performed as described in [\[SWS\\_J1939Tp\\_00097\]](#), and the connection abort reason shall be set to 1 (Already in one or more connection managed sessions and cannot support another)

after the call to `PduR_J1939TpStartOfReception()`, and to 2 (System resources were needed for another task so this connection managed session was terminated) in the other two cases.]

### 7.5.3 N-SDU Transmission

The upper layer (`PDU Router`) asks for the transmission of an N-SDU by calling `J1939Tp_Transmit()`. The parameters of `J1939Tp_Transmit()` describe the Identifier of the N-SDU (`TxPduId`) and a reference to a `PduInfoType` that indicates the full length of the N-SDU to transmit (full Tx N-SDU data length) and a pointer to the payload N-SDU, which may contain `meta data` with `SA`, `DA` and Priority depending on the `meta data` configuration of the N-SDU.

**[SWS\_J1939Tp\_00039]** [When configured, the `transport protocol` variant (`BAM/CMDT`, see `J1939TpTxProtocolType`) and the `SA/DA` (see `J1939TpTxSa/J1939TpTxDa`) shall be used for transmission. For N-SDUs with `meta data`, these parameters are optional. If `SA` or `DA` is not configured, the value provided via the `meta data` shall be used. If the `transport protocol` is not configured, it shall be chosen based on the actual `DA`: `BAM` when `DA` is `0xFF`, `CMDT` otherwise.]

**[SWS\_J1939Tp\_00045]** [The function `J1939Tp_Transmit()` shall use the `TxPduId` and the `SduLength` provided in the `PduInfoType` structure. It shall not use the payload of N-SDUs, only the contained `meta data`.]

**[SWS\_J1939Tp\_00047]** [After a transmission request from the upper layer, the `J1939Tp` module shall initiate the transmission by sending:

- For `CMDT`: a `TP.CM_RTS` frame
- For `BAM`: a `TP.CM_BAM` frame

]

**[SWS\_J1939Tp\_00046]** [For each `TP.DT` frame to be sent, the `J1939Tp` module shall previously call `PduR_J1939TpCopyTxData()` with the following parameters:

- the Identifier of the corresponding N-SDU,
- `PduInfoType` structure with up to 7 bytes as `SduLength`,
- the `retry` parameter, and
- a pointer to a location where the upper layer stores its currently available data.

]

**[SWS\_J1939Tp\_00228]** [When `PduR_J1939TpCopyTxData()` returns `BUFREQ_E_BUSY`, the `J1939Tp` shall retry the call to `PduR_J1939TpCopyTxData()` until the data is available or a timeout occurs.]

**[SWS\_J1939Tp\_00218]** [If [J1939TpTxRetrySupport](#) is disabled, the parameter `retry` of `PduR_J1939TpCopyTxData()` shall always be set to the `NULL_PTR`.]

**[SWS\_J1939Tp\_00219]** [For [BAM](#) transmissions, the parameter `retry` of `PduR_J1939TpCopyTxData()` shall always be set to the `NULL_PTR`.]

**[SWS\_J1939Tp\_00220]** [If [J1939TpTxRetrySupport](#) is enabled, a valid `RetryInfoType` struct shall be provided via the parameter `retry` of `PduR_J1939TpCopyTxData()` during [CMDT](#) transmissions.]

See [Chapter 7.5.4.2](#) for a description how the [SAE J1939 Transport Layer](#) module uses the `RetryInfoType` during [CMDT](#) transmission.

**[SWS\_J1939Tp\_00048]** [The [J1939Tp](#) module shall abort the transmission session like specified in [\[SWS\\_J1939Tp\\_00032\]](#) if any of the following conditions occur:

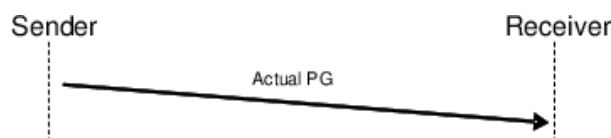
- The upper Layer returns the `PduR_J1939TpCopyTxData()` function call with `BUFREQ_E_NOT_OK` or
- the `J1939Tp_CancelTransmit()` function is called.

In case of a [CMDT](#) connection, a [CMDT Connection Abort](#) shall be performed as described in [\[SWS\\_J1939Tp\\_00097\]](#), and the connection abort reason shall be set to 2 (System resources were needed for another task so this connection managed session was terminated).]

## 7.5.4 Data Flow on the CAN Bus

### 7.5.4.1 Data Flow using Direct Transmission

The following figure shows an example of direct message transmission between two ECUs using [J1939Tp](#). This is the only case of transmission of a [J1939Tp](#) N-SDU using no `TP.CM` or `TP.DT` frame. The `SA` is always included in the CAN identifier. Depending on the [PDU Format](#) of the `PGN`, the CAN Identifier might contain the `DA`.

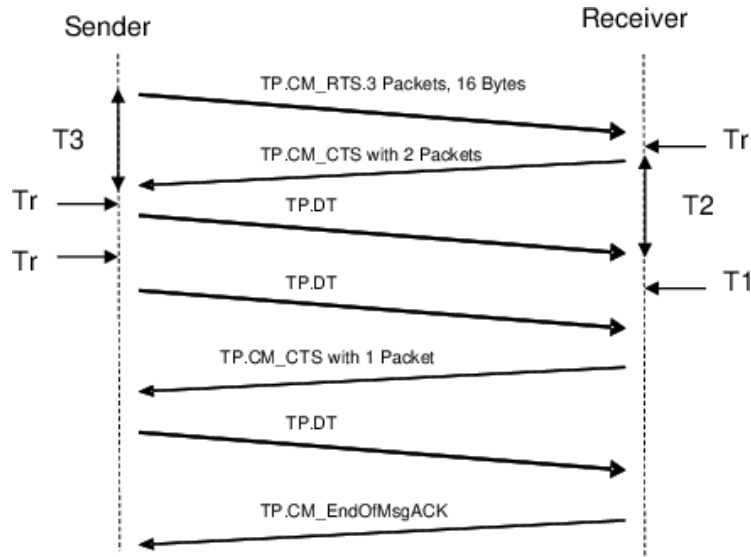


**Figure 7.2: Example of direct data flow for PGs of variable length  $\leq 8$  Bytes**

### 7.5.4.2 Data Flow using CMDT

The following figure shows an example of segmented message transmission between two ECUs using [CMDT](#) as [transport protocol](#) variant. The [CMDT transport protocol](#) variant is used for peer-to-peer communication (i.e. 1 to 1 communication,

like physical addressing in UDS diagnostics). In the example, the transmitted **PG** has a total length of 16 bytes, which corresponds to 3 blocks of 7 bytes.



**Figure 7.3: Example of data flow without error using CDMT as transport protocol variant**

The **J1939 transport protocol** uses the initial sequence of **RTS** and **CTS** to determine the number of packets per block.

**[SWS\_J1939Tp\_00165]** [If **J1939TpTxDynamicBlockCalculation** is enabled, after **J1939TpTransmit()** has been called, the **J1939Tp** shall call **PduR\_J1939TpCopyTxData()** once with **info->SduLength** set to 0 and **retry** set to **NULL\_PTR** to obtain the available amount of data via **availableDataPtr**.]

**[SWS\_J1939Tp\_00207]** [If **J1939TpTxDynamicBlockCalculation** and **J1939TpTxMaxPacketsPerBlock** are enabled, the **J1939Tp** shall compare the available amount of data returned by **PduR\_J1939TpCopyTxData()** to **J1939TpTxMaxPacketsPerBlock** and use the smaller of these two values to calculate the maximum number of packets field of the **TP.CM\_RTS** message.]

**[SWS\_J1939Tp\_00208]** [If **J1939TpTxDynamicBlockCalculation** is enabled but **J1939TpTxMaxPacketsPerBlock** is disabled, the **J1939Tp** shall use the available amount of data returned by **PduR\_J1939TpCopyTxData()** to calculate the maximum number of packets field of the **TP.CM\_RTS** message.]

**[SWS\_J1939Tp\_00209]** [If **J1939TpTxDynamicBlockCalculation** is disabled, the **J1939Tp** shall use **J1939TpTxMaxPacketsPerBlock** for the maximum number of packets field of the **TP.CM\_RTS** message.]

**[SWS\_J1939Tp\_00210]** [If **J1939TpRxDynamicBlockCalculation** and **J1939TpRxPacketsPerBlock** are enabled, the **J1939Tp** shall compare the value returned by **PduR\_J1939TpStartOfReception()** for the available buffer size to **J1939TpRxPacketsPerBlock** and use the lower value to calculate the number of packets field of the **TP.CM\_CTS** message.]

**[SWS\_J1939Tp\_00211]** [If `J1939TpRxDynamicBlockCalculation` is enabled but `J1939TpRxPacketsPerBlock` is disabled, the `J1939Tp` shall use the value returned by `PduR_J1939TpStartOfReception()` for the available buffer size to calculate the number of packets field of the `TP.CM_CTS` message.]

**[SWS\_J1939Tp\_00226]** [After reception of the `TP.CM_RTS` and after reception of the last N-PDU of a block, if the reported buffer size is large enough for the next block, the `J1939Tp` shall transmit a `TP.CM_CTS` message requesting the next block. The number of packets requested by `TP.CM_CTS` shall be constant during the complete reception of one N-SDU, only in the last `TP.CM_CTS` this number shall be reduced to the number of remaining packets.]

**[SWS\_J1939Tp\_00227]** [When there is not enough buffer reported by `PduR_J1939TpStartOfReception()` or `PduR_J1939TpCopyRxData()` for the reception of a complete block, the `J1939Tp` shall call `PduR_J1939TpCopyRxData()` with `info->SduLength` set to 0 until the buffer is large enough for one block, or a timeout occurs.]

Please note: A timeout can be a timeout of an expected message (e.g. T1) or a timeout during transmission of a message (e.g. Tr).

**[SWS\_J1939Tp\_00229]** [While monitoring the buffer state as defined by **[SWS\_J1939Tp\_00227]**, the `J1939Tp` shall send `TP.CM_CTS` wait frames (number of packets set to 0, see also SAE J1939-21) according to the timing requirements defined in SAE J1939-21.]

**[SWS\_J1939Tp\_00212]** [If `J1939TpRxRetrySupport` is enabled, the `J1939Tp` shall adapt the value returned by `PduR_J1939TpStartOfReception()` for the available buffer size according to `J1939TpRxDynamicBufferRatio` before using it to calculate the number of packets field of the `TP.CM_CTS` message as specified in **[SWS\_J1939Tp\_00210]** and **[SWS\_J1939Tp\_00211]**.]

**[SWS\_J1939Tp\_00213]** [If `J1939TpRxDynamicBlockCalculation` is disabled, the `J1939Tp` shall use `J1939TpRxPacketsPerBlock` to calculate the number of packets field of the `TP.CM_CTS` message.]

**[SWS\_J1939Tp\_00217]** [If `J1939TpTxRetrySupport` is enabled, the `J1939Tp` shall call `PduR_J1939TpCopyTxData()` with

- `TpDataState` set to `TP_DATACONF` for the first call after reception of a `TP.CM_CTS`, and
- `TpDataState` set to `TP_CONFENDING` for the following calls.

]

**[SWS\_J1939Tp\_00195]** [If a `TP.CM_CTS` wait frame (number of packets set to 0, see also SAE J1939-21) is received, the `J1939Tp` shall wait for another `TP.CM_CTS` frame.]



**[SWS\_J1939Tp\_00223]** [When the `J1939Tp` receives a `TP.CM_CTS` frame that requests data beyond the current position or preceding the position where the last `TP.CM_CTS` was received, it shall abort the transmission using the mechanisms described by **[SWS\_J1939Tp\_00032]** and **[SWS\_J1939Tp\_00097]** with reason 255 (`SNA`).]

**[SWS\_J1939Tp\_00221]** [If `J1939TpTxRetrySupport` is enabled, when the `J1939Tp` receives a `TP.CM_CTS` frame requesting already sent data, it shall call `PduR_J1939TpCopyTxData()` with `TpDataState` set to `TP_DATA_RETRY` and `TxTpDataCnt` set to the number of bytes to be retransmitted.]

The number of bytes that need to be retransmitted is calculated from the position of the requested package relative to the current package.

**[SWS\_J1939Tp\_00194]** [If `J1939TpTxRetrySupport` is disabled, when the `J1939Tp` receives a `TP.CM_CTS` frame requesting already sent data, it shall abort the transmission using the mechanisms described by **[SWS\_J1939Tp\_00032]** and **[SWS\_J1939Tp\_00097]** with reason 255 (unassigned).]

**[SWS\_J1939Tp\_00222]** [If `J1939TpRxRetrySupport` is enabled, when a sequence error occurs during reception, the `J1939Tp` module shall, after the last `TP.DT` message of the block was received, send a `TP.CM_CTS` frame requesting the packages that follow the last correctly received package of the current block.]

**[SWS\_J1939Tp\_00232]** [The content of a `TP.DT` message with a sequence error and of the `TP.DT` messages following a sequence error in the same block shall be discarded.]

Note: This means that for such messages `PduR_J1939TpCopyRxData()` shall not be called.

**[SWS\_J1939Tp\_00216]** [If `J1939TpRxRetrySupport` is disabled, when a sequence error occurs during reception, the `J1939Tp` module shall abort the reception session

- as specified in **[SWS\_J1939Tp\_00031]** and
- as specified in **[SWS\_J1939Tp\_00097]** with connection abort reason 255 (`SNA`).

]

**[SWS\_J1939Tp\_00123]**

*Upstream requirements:* **[SRS\_J1939\_00022]**

[`J1939Tp` shall implement all `CMDT` related timing constraints (`Tr`, `Th`, `T1`, `T2`, `T3`, `T4`) as described in **SAE J1939-21**. They supervise the `CMDT` data flow.]

**[SWS\_J1939Tp\_00100]** [If a timeout occurs during `CMDT` transmission (see **SAE J1939-21** for details) then the `J1939Tp` module shall abort the transmission session

- as specified in **[SWS\_J1939Tp\_00032]** and

- as specified in [SWS\_J1939Tp\_00097] with connection abort reason 3 (a timeout occurred and this is the connection abort to close the session) when the timeout occurred after successful transmission of the TP.CM\_RTS frame.

]

**[SWS\_J1939Tp\_00159]** [If a timeout occurs during reception (see SAE J1939-21 for details) then the J1939Tp module shall abort the reception session

- as specified in [SWS\_J1939Tp\_00031] and
- as specified in [SWS\_J1939Tp\_00097] with connection abort reason 3 (A timeout occurred and this is the connection abort to close the session).

]

**[SWS\_J1939Tp\_00098]** [If a TP.Conn\_Abort frame is received after a TP.CM\_RTS frame has been successfully sent by the ECU, the transmission shall be aborted like indicated in [SWS\_J1939Tp\_00032]. No transmission of TP.Conn\_Abort is necessary.]

**[SWS\_J1939Tp\_00163]** [If a TP.Conn\_Abort frame is received after a TP.CM\_RTS frame has been received, the reception shall be aborted like indicated in [SWS\_J1939Tp\_00031]. No transmission of TP.Conn\_Abort is necessary.]

### 7.5.4.3 Data Flow using BAM

#### **[SWS\_J1939Tp\_00121]**

*Upstream requirements:* SRS\_J1939\_00022

[J1939Tp shall implement the BAM related timing constraints (50ms or J1939TpTxBamTiming, Tr, T1) as described in SAE J1939-21. They supervise the BAM data flow.]

**[SWS\_J1939Tp\_00160]** [If a timeout occurs during the BAM reception, the J1939Tp module shall abort the reception session as specified in [SWS\_J1939Tp\_00031].]

**[SWS\_J1939Tp\_00192]** [After a sequence error, J1939Tp shall abort BAM connections as specified in [SWS\_J1939Tp\_00031].]

The following figure shows an example of segmented message transmission between two ECUs using BAM as transport protocol variant according to SAE J1939-21. The BAM transport protocol variant is used for a broadcast communication (i.e. 1 to n communication, like functional addressing in UDS diagnostics). In the example, the transmitted PG has a total length of 3 blocks of 7 bytes that have to be successively transmitted.



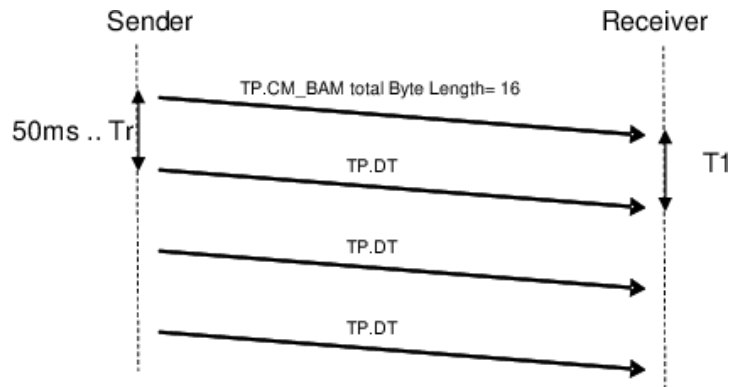


Figure 7.4: Example of data flow using BAM as transport protocol variant

### 7.5.5 N-SDU Buffer Management

**J1939Tp** shall have no internal PDU buffers. It requests data for sending directly from the upper layers via `PduR_J1939TpCopyTxData()` and provides received data directly to the upper layers via `PduR_J1939TpCopyRxData()`.

To guarantee data consistency, the complete buffer of the upper layers must be locked during the whole data transmission or reception.

Data transmission is initialized when the **PduR** calls `J1939Tp_Transmit()` and is active after **J1939Tp** executed this call successfully until **J1939Tp** calls `PduR_J1939TpTxConfirmation()`:

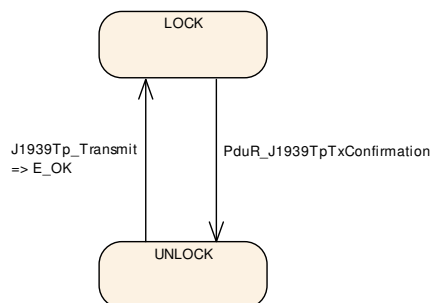
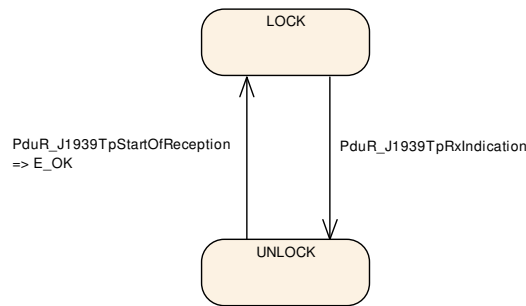


Figure 7.5: Tx Buffer locking

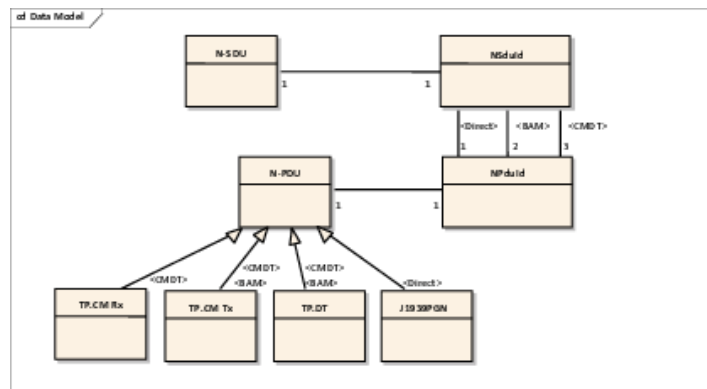
Data reception is initialized when **J1939Tp** calls `PduR_J1939TpStartOfReception()` and is active after the **PduR** executed this call successfully until **J1939Tp** calls `PduR_J1939TpRxIndication()`:



**Figure 7.6: Rx Buffer locking**

### 7.5.6 Relationship between N-SDU and N-PDU in J1939Tp

This subsection describes the relation that exists between an N-SDU and the set of N-PDUs that is required to transport the N-SDU data, as shown in the following figure.



**Figure 7.7: Relation between N-SDU and N-PDU**

The N-PDUs as well as the N-SDU may use [meta data](#). In that case, the N-SDU stands for a certain [PGN](#), and the N-PDUs stand for [TP.CM](#), [TP.DT](#), and direct [PG](#), regardless of source and/or [destination addresses](#).

**[SWS\_J1939Tp\_00057]** [During reception, the N-SDU is identified by a combination of the [PGN](#) included in the payload of [TP.CM](#) and, when the N-SDU has no [meta data](#), from the addressing information implicitly encoded in the NPduids or explicitly provided via the [meta data](#) of the N-PDUs.]

**[SWS\_J1939Tp\_00199]** [During transmission, the relevant set of N-PDUs is identified by the configured [SA/DA](#) of N-SDUs without [meta data](#), or by the [SA/DA](#) explicitly provided in the [meta data](#) by the upper layer.]

### 7.5.7 Concurrent Connections

Connections only concern internal J1939Tp purposes. They are transparent for the upper and lower layers of J1939Tp but influence the handling of J1939Tp.

A J1939Tp connection is characterized by its direction (Receiving /Sending) and its type (BAM / CMDT / Direct). A J1939Tp connection of type BAM or CMDT uses the following transport related frames:

- control (TP.CM)
- data (TP.DT)

The CAN identifier corresponding to those transport specific frames is the same for all J1939 PGs longer than 8 bytes that are

- sent from a given SA in the case of BAM, or
- sent from a given SA to a given DA in the case of CMDT.

This constrains the possibility for J1939Tp to process concurrent connections.

**[SWS\_J1939Tp\_00120]** [The J1939Tp shall be able to handle connections in parallel for all N-SDUs that do not interfere in the usage of TP.DT frames with the same SA and DA. For channels with defined SA/DA and protocol type, only one TP connection and one direct connection for each PG can be open at any time. For channels with variable SA/DA, the maximum number of parallel connections is limited by the number of N-SDUs assigned to this channel.]

J1939Tp shall only support concurrent connections as described in subsection 5.10.5 of SAE J1939-21. Note that one AUTOSAR ECU can represent several J1939 nodes and thus may have more than one address (used as SA or DA) assigned to it.

#### **[SWS\_J1939Tp\_00062]**

*Upstream requirements:* SRS\_J1939\_00025

[Each connection shall be independent of the other connections.]

This means that a connection shall use its own resources, such as timer or state machine.

#### **[SWS\_J1939Tp\_00237] Deferred Transmission**

*Upstream requirements:* SRS\_J1939\_00067

[If the J1939Tp cannot start a transmission because the N-SDU interferes with another currently transmitted N-SDU due to identical SA/DA and protocol type, the transmission of the TP.CM\_RTS or TP.CM\_BAM frame shall be deferred until the interfering N-SDU transmission is finished.]

**[SWS\_J1939Tp\_00238] Prioritization of Deferred Transmission**

*Upstream requirements:* [SRS\\_J1939\\_00067](#)

[If several deferred N-SDUs interfere with each other because of identical [SA/DA](#) and protocol type, a numerically lower [PGN](#) shall take precedence.]

**7.5.8 N-PDU Padding**

**[SWS\_J1939Tp\_00200]** [The [SAE J1939 Transport Layer](#) module shall send [TP.DT](#) frames always with 8 bytes according to [SAE J1939-21](#).]

**[SWS\_J1939Tp\_00068]**

*Upstream requirements:* [SRS\\_J1939\\_00024](#)

[According to [SAE J1939-21](#), all unused data bytes within the last [TP.DT](#) frame or the direct frame shall be set to 0xFF ([SNA](#)).]

**7.5.9 Handling of Unexpected N-PDU Arrivals****[SWS\_J1939Tp\_00064]**

*Upstream requirements:* [SRS\\_J1939\\_00023](#)

[The [J1939Tp](#) shall ignore unexpected N-PDUs that do not correspond to a currently active connection.]

**[SWS\_J1939Tp\_00224]** [If [J1939TpTxRetrySupport](#) is disabled, when a [TP.CM\\_CTS](#) is received while [TP.DT](#) messages are being transmitted, [J1939Tp](#) shall abort the connection as specified in [\[SWS\\_J1939Tp\\_00032\]](#) and in [\[SWS\\_J1939Tp\\_00097\]](#) with connection abort reason 4 ([CTS](#) messages received when data transfer is in progress).]

**[SWS\_J1939Tp\_00225]** [When a [TP.CM\\_RTS](#) is received for a currently active connection, the [J1939Tp](#) shall stop this connection as specified in [\[SWS\\_J1939Tp\\_00031\]](#) and start a new connection as described in [\[SWS\\_J1939Tp\\_00038\]](#).]

**7.6 Error Classification**

Chapter [\[16, SWS BSW General\]](#) 7.2 “*Error Handling*” describes the error handling of the [Basic Software](#) in detail. Above all, it constitutes a classification scheme consisting of five error types which may occur in [BSW](#) modules.

Based on this foundation, this section specifies particular errors arranged in the respective subsections below.

## 7.6.1 Development Errors

### [SWS\_J1939Tp\_00115] Definition of development errors in module J1939Tp

Upstream requirements: [SRS\\_BSW\\_00327](#), [SRS\\_BSW\\_00337](#), [SRS\\_BSW\\_00385](#), [SRS\\_BSW\\_00441](#)

[

Type of error	Related error code	Error value
API service used in state J1939TP_OFF	J1939TP_E_UNINIT	0x01
J1939Tp_Init() called in state J1939TP_ON.	J1939TP_E_REINIT	0x02
J1939Tp_Init() was called with an invalid configuration pointer	J1939TP_E_INIT_FAILED	0x03
API service called with null pointer.	J1939TP_E_PARAM_POINTER	0x10
API service called with wrong ID.	J1939TP_E_INVALID_PDU_SDU_ID	0x11

]

## 7.6.2 Runtime Errors

### [SWS\_J1939Tp\_00234] Definition of runtime errors in module J1939Tp [

Type of error	Related error code	Error value
Timeout occurred on receiver side after reception of an intermediate TP.DT frame of a block.	J1939TP_E_TIMEOUT_T1	0x30
Timeout occurred on receiver side after transmission of a TP.CM/CTS frame.	J1939TP_E_TIMEOUT_T2	0x31
Timeout occurred on transmitter side after transmission of the last TP.DT frame of a block.	J1939TP_E_TIMEOUT_T3	0x32
Timeout occurred on transmitter side after reception of a TP.CM/CTS(0) frame.	J1939TP_E_TIMEOUT_T4	0x33
Timeout occurred on transmitter or receiver side while trying to send the next TP.DT or TP.CM frame.	J1939TP_E_TIMEOUT_TR	0x34
Timeout occurred on receiver side while trying to send the next TP.CM/CTS frame after a TP.CM/CTS(0) frame.	J1939TP_E_TIMEOUT_TH	0x35
Invalid value for "total message size" in received TP.CM/RTS frame.	J1939TP_E_INVALID_TMS	0x40
Value for "total number of packets" in received TP.CM/RTS frame does not match the "total message size".	J1939TP_E_INVALID_TNOP	0x41
Invalid value for "maximum number of packets" in received TP.CM/RTS frame.	J1939TP_E_INVALID_MNOP	0x42
Unexpected PGN in received TP.CM frame.	J1939TP_E_INVALID_PGN	0x43
Invalid value for "number of packets" in received TP.CM/CTS frame.	J1939TP_E_INVALID_NOP	0x44
Invalid value for "next packet number" in received TP.CM/CTS frame.	J1939TP_E_INVALID_NPN	0x45
Invalid value for "connection abort reason" in received TP.Conn_Abort frame.	J1939TP_E_INVALID_CAR	0x46
Unexpected serial number in received TP.DT frame.	J1939TP_E_INVALID_SN	0x47

]

### **7.6.3 Production Errors**

There are no production errors.

### **7.6.4 Extended Production Errors**

There are no extended production errors.

## 8 API Specification

### 8.1 API Parameter Checking

[J1939TP\\_E\\_PARAM\\_POINTER](#) shall be reported as specified in [16, SWS BSW General] by [SWS\_BSW\_00212].

#### [SWS\_J1939Tp\_00188]

*Upstream requirements:* [SRS\\_BSW\\_00323](#), [SRS\\_BSW\\_00386](#)

[If development error detection for the [J1939Tp](#) is enabled via [J1939TpDevErrorDetect](#), all APIs using a SDU- or PDU-Identifier shall check the input Identifier and raise the development error [J1939TP\\_E\\_INVALID\\_PDU\\_SDU\\_ID](#) in case the API has been called for a not configured PDU or SDU.]

### 8.2 Imported Types

This section lists all externally defined types that are used by [J1939Tp](#). These types are included from the headers corresponding to the module names listed in the table below.

#### [SWS\_J1939Tp\_00230] Definition of imported datatypes of module J1939Tp [

Module	Header File	Imported Type
Comtype	ComStack_Types.h	BufReq_ReturnType
	ComStack_Types.h	PduIdType
	ComStack_Types.h	PduInfoType
	ComStack_Types.h	PduLengthType
	ComStack_Types.h	RetryInfoType
	ComStack_Types.h	TpDataStateType
Std	Std_Types.h	Std_ReturnType
	Std_Types.h	Std_VersionInfoType

]

### 8.3 Type Definitions

This section lists the types defined by [J1939Tp](#).

#### [SWS\_J1939Tp\_00175] Definition of datatype J1939Tp\_ConfigType [

Name	J1939Tp_ConfigType
Kind	Structure
Elements	implementation specific



△

	<b>Type</b>	–
	<b>Comment</b>	The content of the initialization data structure is implementation specific.
<b>Description</b>	Data structure containing post-build configuration data of J1939-TP.	
<b>Available via</b>	J1939Tp.h	

]

The `J1939Tp_ConfigType` defines a structure that contains configuration parameters `J1939Tp` uses at run time. It is provided as an argument to `J1939Tp_Init()`.

## 8.4 Function Definitions

This section defines a list of functions provided for upper layer modules.

### 8.4.1 J1939Tp\_Init

#### [SWS\_J1939Tp\_00087] Definition of API function J1939Tp\_Init

Upstream requirements: [SRS\\_BSW\\_00101](#), [SRS\\_BSW\\_00358](#)

[

<b>Service Name</b>	J1939Tp_Init	
<b>Syntax</b>	<pre>void J1939Tp_Init (     const J1939Tp_ConfigType* ConfigPtr )</pre>	
<b>Service ID [hex]</b>	0x01	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Non Reentrant	
<b>Parameters (in)</b>	ConfigPtr	Pointer to configuration data structure.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	None	
<b>Description</b>	This function initializes the J1939Tp module.	
<b>Available via</b>	J1939Tp.h	

]

After power up, `J1939Tp` is in a state called `J1939TP_OFF`. In this state, `J1939Tp` is not yet configured, and therefore cannot perform any communication task. The `J1939Tp` module's environment (usually the ECU Manager) will call `J1939Tp_Init()` before using the `J1939Tp` module for further processing.



**[SWS\_J1939Tp\_00024]**

*Upstream requirements:* [SRS\\_J1939\\_00010](#)

[The function [J1939Tp\\_Init\(\)](#) shall initialize all global variables of the module and reset all [transport protocol](#) connections.]

**[SWS\_J1939Tp\_00022]** [[J1939Tp\\_Init\(\)](#) shall change to the internal state [J1939TP\\_ON](#) after successful initialization.]

**[SWS\_J1939Tp\_00026]**

*Upstream requirements:* [SRS\\_BSW\\_00386](#)

[If called when the [J1939Tp](#) module is in the global state [J1939TP\\_ON](#), the function [J1939Tp\\_Init\(\)](#) shall raise the development error [J1939TP\\_E\\_REINIT](#) and do nothing.]

**[SWS\_J1939Tp\_00187]**

*Upstream requirements:* [SRS\\_BSW\\_00400](#), [SRS\\_BSW\\_00405](#), [SRS\\_BSW\\_00414](#), [SRS\\_BSW\\_00438](#)

[The provided [ConfigPtr](#) shall only be used, when post-build configuration is enabled, or when different configuration variants must be supported. Otherwise, the parameters should be accessed directly to avoid indirection via the [ConfigPtr](#).]

The structure of type [J1939Tp\\_ConfigType](#) pointed to by the [ConfigPtr](#) contains post-build parameters of the [J1939Tp](#) module. In link time or pre-compile configured environments, the structure may contain a set of post-build parameters that differ between several configuration variants. It is expected that link time and pre-compile parameters may not change for different configuration variants.

[J1939Tp\\_Init\(\)](#) has no return value because configuration data errors should be detected during configuration time (e.g. by the configuration tools). Furthermore, if a hardware error occurs, it will be reported via the error manager modules.

[J1939TP\\_E\\_INIT\\_FAILED](#) shall be reported as specified in [16, SWS BSW General] by [\[SWS\\_BSW\\_00050\]](#).

### 8.4.2 J1939Tp\_Shutdown

#### [SWS\_J1939Tp\_00093] Definition of API function J1939Tp\_Shutdown

Upstream requirements: [SRS\\_BSW\\_00336](#)

[

<b>Service Name</b>	J1939Tp_Shutdown
<b>Syntax</b>	<pre>void J1939Tp_Shutdown (     void )</pre>
<b>Service ID [hex]</b>	0x02
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Non Reentrant
<b>Parameters (in)</b>	None
<b>Parameters (inout)</b>	None
<b>Parameters (out)</b>	None
<b>Return value</b>	None
<b>Description</b>	This function is used to shutdown the J1939Tp module.
<b>Available via</b>	J1939Tp.h

]

To close down communication, the state handling (usually the ECU Manager) calls [J1939Tp\\_Shutdown\(\)](#).

[SWS\_J1939Tp\_00094] [[J1939Tp\\_Shutdown\(\)](#) shall close all pending [transport protocol](#) connections, free all resources and set the [J1939Tp](#) module into the global state `J1939TP_OFF` state.]

[SWS\_J1939Tp\_00095] [[J1939Tp\\_Shutdown\(\)](#) shall not raise a notification about the pending frame transmission or reception.]

### 8.4.3 J1939Tp\_GetVersionInfo

#### [SWS\_J1939Tp\_00089] Definition of API function J1939Tp\_GetVersionInfo

Upstream requirements: [SRS\\_BSW\\_00407](#)

[

<b>Service Name</b>	J1939Tp_GetVersionInfo
<b>Syntax</b>	<pre>void J1939Tp_GetVersionInfo (     Std_VersionInfoType* VersionInfo )</pre>
<b>Service ID [hex]</b>	0x03
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Reentrant
<b>Parameters (in)</b>	None

▽



<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	VersionInfo	Pointer to the location where the version information of J1939Tp shall be stored.
<b>Return value</b>	None	
<b>Description</b>	Returns the version information of J1939Tp.	
<b>Available via</b>	J1939Tp.h	

]

See [16, SWS BSW General] 8.3.4 “*Get Version Information*” for details. The module ID of the [SAE J1939 Transport Layer](#) is defined in [16, SWS BSW General] 11.2 “*Modules*”.

Note that the function [J1939Tp\\_GetVersionInfo\(\)](#) may be called in global state J1939TP\_OFF, i.e. before initialization of the [J1939Tp](#) module.

**[SWS\_J1939Tp\_00235]** [The [J1939Tp\\_GetVersionInfo\(\)](#) API shall only be available when [J1939TpVersionInfoApi](#) is enabled.]

#### 8.4.4 J1939Tp\_Transmit

##### [SWS\_J1939Tp\_00096] Definition of API function J1939Tp\_Transmit

Upstream requirements: [SRS\\_BSW\\_00357](#)

[

<b>Service Name</b>	J1939Tp_Transmit	
<b>Syntax</b>	<pre>Std_ReturnType J1939Tp_Transmit (     PduIdType TxPduId,     const PduInfoType* PduInfoPtr )</pre>	
<b>Service ID [hex]</b>	0x49	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Reentrant for different PduIds. Non reentrant for the same PduId.	
<b>Parameters (in)</b>	TxPduId	Identifier of the PDU to be transmitted
	PduInfoPtr	Length of and pointer to the PDU data and pointer to MetaData.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	Std_ReturnType	E_OK: Transmit request has been accepted. E_NOT_OK: Transmit request has not been accepted.
<b>Description</b>	Requests transmission of a PDU.	
<b>Available via</b>	J1939Tp.h	

]

As described in [SWS\_J1939Tp\_00119] and [SWS\_J1939Tp\_00032], the [J1939Tp](#) module will notify the upper layer by calling the [PduR\\_J1939TpTxConfirmation\(\)](#) callback when the transmit request has been completed.

[SWS\_J1939Tp\_00101] [The function `J1939Tp_Transmit()` shall reject a request, if the `J1939Tp_Transmit()` service is called for an N-SDU identifier that is being used in a currently running `SAE J1939 Transport Layer` session.]

#### 8.4.5 J1939Tp\_CancelTransmit

[SWS\_J1939Tp\_00177] Definition of API function `J1939Tp_CancelTransmit` [

<b>Service Name</b>	J1939Tp_CancelTransmit	
<b>Syntax</b>	Std_ReturnType J1939Tp_CancelTransmit ( PduIdType TxPduId )	
<b>Service ID [hex]</b>	0x4a	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Reentrant for different PduIds. Non reentrant for the same PduId.	
<b>Parameters (in)</b>	TxPduId	Identification of the PDU to be cancelled.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	Std_ReturnType	E_OK: Cancellation was executed successfully by the destination module. E_NOT_OK: Cancellation was rejected by the destination module.
<b>Description</b>	Requests cancellation of an ongoing transmission of a PDU in a lower layer communication module.	
<b>Available via</b>	J1939Tp.h	

]

[SWS\_J1939Tp\_00203] [`J1939Tp_CancelTransmit()` shall return `E_NOT_OK` if `TxPduId` is invalid, if `TxPduId` is currently not active, if currently a direct frame is transmitted, if the last `TP.DT` frame has already been transmitted during `BAM` transmission, or if the `TP.CM_EOMAck` frame has already been received during `CMDT` transmission.]

[SWS\_J1939Tp\_00214] [The `J1939Tp_CancelTransmit()` API shall only be available when `J1939TpCancellationSupport` is enabled.]

#### 8.4.6 J1939Tp\_CancelReceive

[SWS\_J1939Tp\_00176] Definition of API function `J1939Tp_CancelReceive` [

<b>Service Name</b>	J1939Tp_CancelReceive	
<b>Syntax</b>	Std_ReturnType J1939Tp_CancelReceive ( PduIdType RxPduId )	
<b>Service ID [hex]</b>	0x4c	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Non Reentrant	





<b>Parameters (in)</b>	RxPduId	Identification of the PDU to be cancelled.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	Std_ReturnType	E_OK: Cancellation was executed successfully by the destination module. E_NOT_OK: Cancellation was rejected by the destination module.
<b>Description</b>	Requests cancellation of an ongoing reception of a PDU in a lower layer transport protocol module.	
<b>Available via</b>	J1939Tp.h	

]

**[SWS\_J1939Tp\_00204]** [J1939Tp\_CancelReceive() shall return E\_NOT\_OK if RxPduId is invalid, if RxPduId is currently not active, if currently a direct frame is received, if the last TP.DT frame has already been received during BAM reception, or if the TP.CM\_EOMAck frame has already been sent during CMDT reception.]

**[SWS\_J1939Tp\_00215]** [The J1939Tp\_CancelReceive() API shall only be available when J1939TpCancellationSupport is enabled.]

## 8.5 Callback Notifications

This is a list of functions provided for other modules.

### 8.5.1 J1939Tp\_RxIndication

#### [SWS\_J1939Tp\_00108] Definition of callback function J1939Tp\_RxIndication

Upstream requirements: SRS\_BSW\_00359, SRS\_BSW\_00360

[

<b>Service Name</b>	J1939Tp_RxIndication	
<b>Syntax</b>	<pre>void J1939Tp_RxIndication (     PduIdType RxPduId,     const PduInfoType* PduInfoPtr )</pre>	
<b>Service ID [hex]</b>	0x42	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Reentrant for different PduIds. Non reentrant for the same PduId.	
<b>Parameters (in)</b>	RxPduId	ID of the received PDU.
	PduInfoPtr	Contains the length (SduLength) of the received PDU, a pointer to a buffer (SduDataPtr) containing the PDU, and the MetaData related to this PDU.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	None	





<b>Description</b>	Indication of a received PDU from a lower layer communication interface module.
<b>Available via</b>	J1939Tp.h

]

The `J1939Tp` module provides the `J1939Tp_RxIndication()` API to allow the `CanIf` to notify via the `LSduR` that a new N-PDU has been received.

#### [SWS\_J1939Tp\_00110]

*Upstream requirements:* [SRS\\_BSW\\_00333](#)

[The function `J1939Tp_RxIndication()` shall be callable in interrupt context (it could be called from the CAN receive interrupt).]

### 8.5.2 J1939Tp\_TxConfirmation

#### [SWS\_J1939Tp\_00112] Definition of callback function J1939Tp\_TxConfirmation

*Upstream requirements:* [SRS\\_BSW\\_00359](#), [SRS\\_BSW\\_00360](#)

[

<b>Service Name</b>	J1939Tp_TxConfirmation	
<b>Syntax</b>	<pre>void J1939Tp_TxConfirmation (     PduIdType TxPduId,     Std_ReturnType result )</pre>	
<b>Service ID [hex]</b>	0x40	
<b>Sync/Async</b>	Synchronous	
<b>Reentrancy</b>	Reentrant for different PduIds. Non reentrant for the same PduId.	
<b>Parameters (in)</b>	TxPduId	ID of the PDU that has been transmitted.
	result	E_OK: The PDU was transmitted. E_NOT_OK: Transmission of the PDU failed.
<b>Parameters (inout)</b>	None	
<b>Parameters (out)</b>	None	
<b>Return value</b>	None	
<b>Description</b>	The lower layer communication interface module confirms the transmission of a PDU, or the failure to transmit a PDU.	
<b>Available via</b>	J1939Tp.h	

]

The `J1939Tp` module implements the `J1939Tp_TxConfirmation()` API to allow the `CanIf` module to confirm via the `LSduR` that a TP related frame (`TP.CM`, `TP.DT`) or direct frame has been successfully transmitted to the J1939 network.

#### [SWS\_J1939Tp\_00114]

*Upstream requirements:* [SRS\\_BSW\\_00333](#)

[The function `J1939Tp_TxConfirmation()` shall be callable in interrupt context (it could be called from the CAN transmit interrupt).]

## 8.6 Scheduled Functions

The Basic Software Scheduler directly calls the functions listed in this section. Scheduled functions shall have no return value and no parameter, and need not be reentrant.

### 8.6.1 J1939Tp\_MainFunction

#### [SWS\_J1939Tp\_00104] Definition of scheduled function J1939Tp\_MainFunction

Upstream requirements: [SRS\\_BSW\\_00373](#)

[

<b>Service Name</b>	J1939Tp_MainFunction
<b>Syntax</b>	void J1939Tp_MainFunction ( void )
<b>Service ID [hex]</b>	0x04
<b>Description</b>	Main function of the J1939Tp. Used for scheduling purposes and timeout supervision.
<b>Available via</b>	SchM_J1939Tp.h

]

#### [SWS\_J1939Tp\_00106]

Upstream requirements: [SRS\\_BSW\\_00478](#)

[The calling frequency of the function [J1939Tp\\_MainFunction\(\)](#) is determined by the configuration parameter [J1939TpMainFunctionPeriod](#).]

## 8.7 Expected Interfaces

In this section all interfaces required from other modules are listed.

### 8.7.1 Mandatory Interfaces

This subsection defines all interfaces, which are required to fulfill the core functionality of the module.

#### [SWS\_J1939Tp\_00116] Definition of mandatory interfaces required by module J1939Tp [

API Function	Header File	Description
Det_ReportRuntimeError	Det.h	Service to report runtime errors. If a callout has been configured then this callout shall be called.
LSduR_J1939TpTransmit (draft)	LSduR_J1939Tp.h	Requests transmission of a PDU.

▽



API Function	Header File	Description
PduR_J1939TpCopyRxData	PduR_J1939Tp.h	This function is called to provide the received data of an I-PDU segment (N-PDU) to the upper layer. Each call to this function provides the next part of the I-PDU data. The size of the remaining buffer is written to the position indicated by bufferSizePtr.
PduR_J1939TpCopyTxData	PduR_J1939Tp.h	This function is called to acquire the transmit data of an I-PDU segment (N-PDU). Each call to this function provides the next part of the I-PDU data unless retry->TpDataState is TP_DATA_RETRY. In this case the function restarts to copy the data beginning at the offset from the current position indicated by retry->TxTpDataCnt. The size of the remaining data is written to the position indicated by availableDataPtr.
PduR_J1939TpRxIndication	PduR_J1939Tp.h	Called after an I-PDU has been received via the TP API, the result indicates whether the transmission was successful or not.
PduR_J1939TpStartOfReception	PduR_J1939Tp.h	This function is called at the start of receiving an N-SDU. The N-SDU might be fragmented into multiple N-PDUs (FF with one or more following CFs) or might consist of a single N-PDU (SF). The service shall provide the currently available maximum buffer size when invoked with TpSdu Length equal to 0.
PduR_J1939TpTxConfirmation	PduR_J1939Tp.h	This function is called after the I-PDU has been transmitted on its network, the result indicates whether the transmission was successful or not.

## 8.7.2 Optional Interfaces

This subsection defines all interfaces that are required to fulfill an optional functionality of the module.

### [SWS\_J1939Tp\_00060] Definition of optional interfaces requested by module J1939Tp

API Function	Header File	Description
Det_ReportError	Det.h	Service to report development errors.



## 9 Sequence Diagrams

The following sequence diagrams shall give an impression of the way the [SAE J1939 Transport Layer](#) module shall behave and interoperate with other [BSW](#) modules. They are not complete and not binding for the implementation.

### 9.1 Reception of Direct PG

The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [L-SDU Router](#) and the [PDU Router](#) during reception of a direct PG, i.e. an N-SDU with dynamic length that is not larger than 8 bytes.

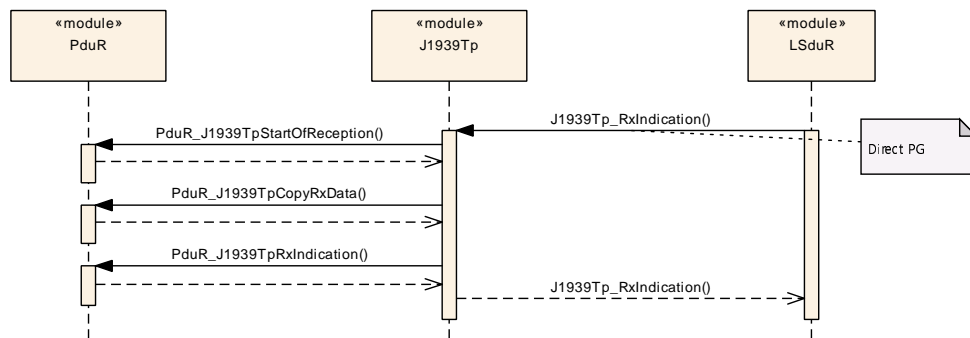


Figure 9.1: Reception of Direct PG

### 9.2 Reception via BAM

The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [L-SDU Router](#) and the [PDU Router](#) during reception of a PG via [BAM](#), i.e. an N-SDU that is larger than 8 bytes and is sent to the whole network.

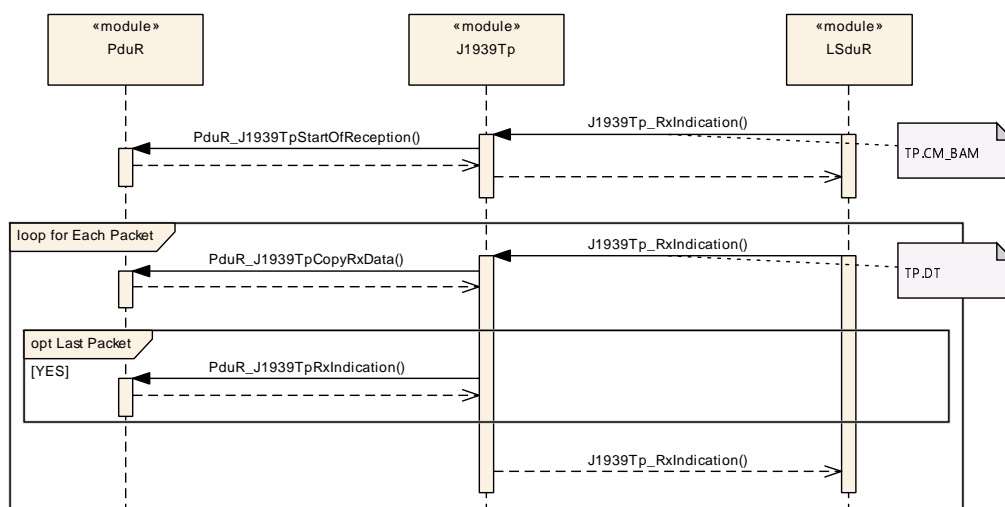


Figure 9.2: Reception via BAM

### 9.3 Reception via CMTD

The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [L-SDU Router](#) and the [PDU Router](#) during reception of a [PG](#) via [CMTD](#), i.e. an N-SDU that is larger than 8 bytes and is sent directly to the receiving node.

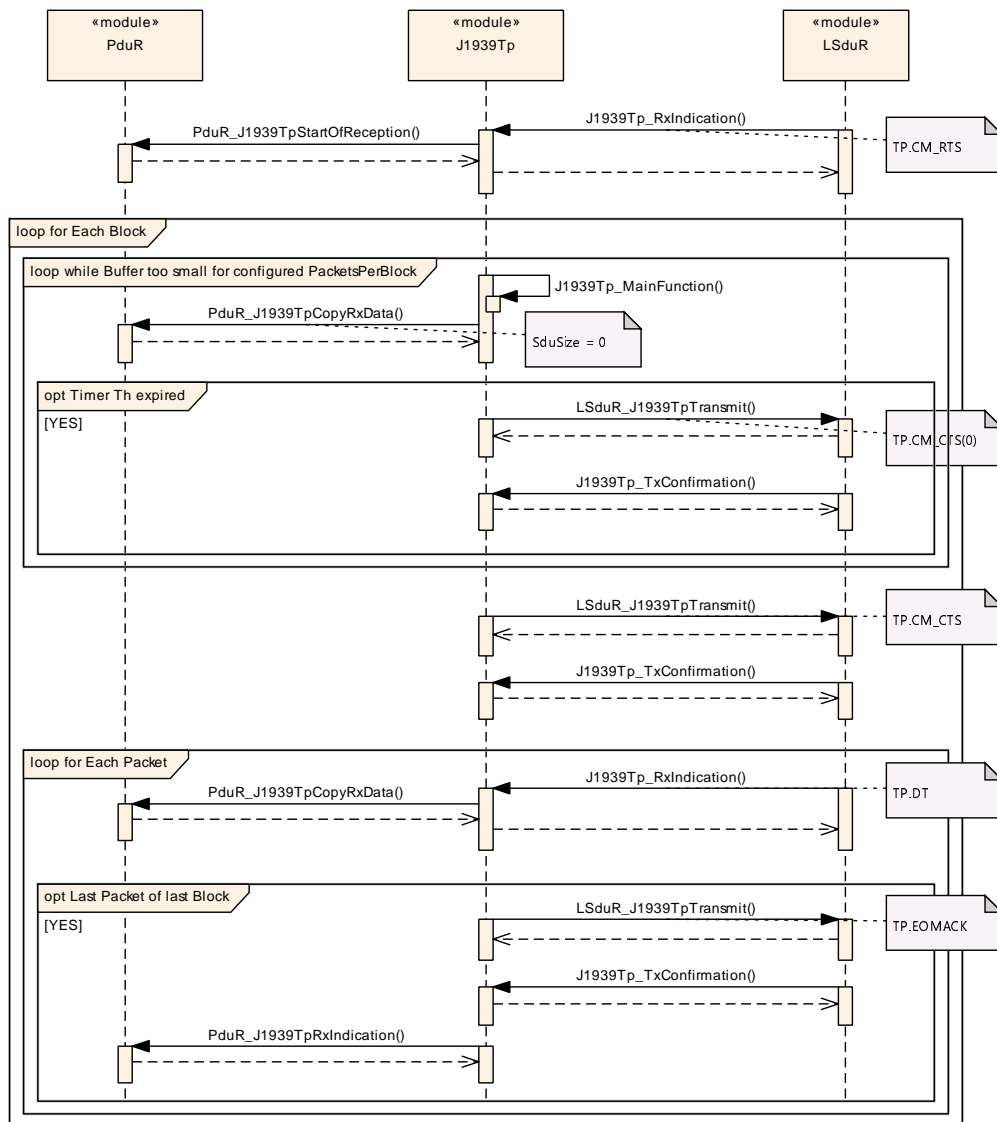
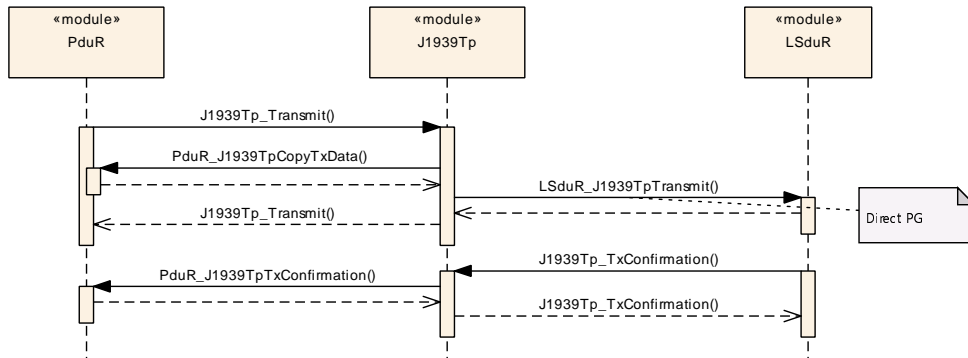


Figure 9.3: Reception via CMTD

### 9.4 Transmission of Direct PG

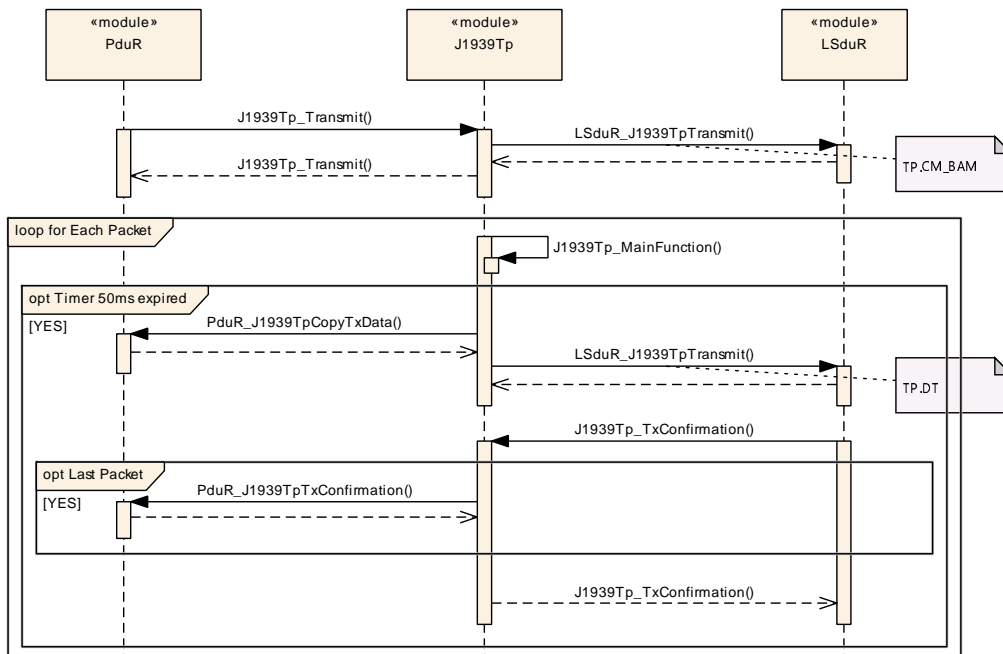
The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [PDU Router](#) and the [L-SDU Router](#) during transmission of a direct [PG](#), i.e. an N-SDU with dynamic length that is not larger than 8 bytes.



**Figure 9.4: Transmission of Direct PG**

## 9.5 Transmission via BAM

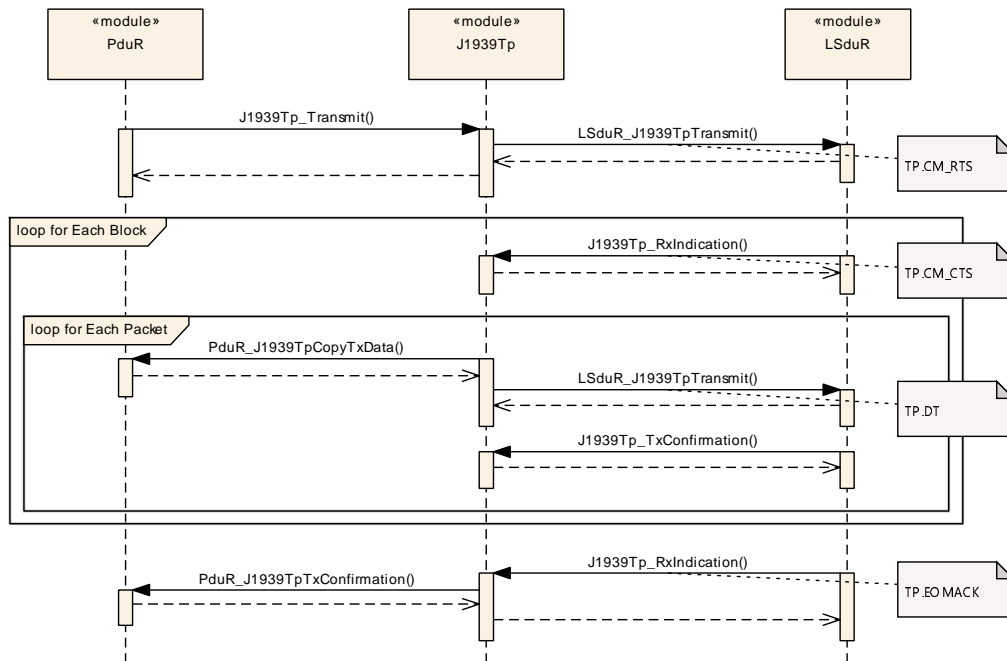
The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [PDU Router](#) and the [L-SDU Router](#) during transmission of a [PG](#) via [BAM](#), i.e. an N-SDU that is larger than 8 bytes and is sent to the whole network.



**Figure 9.5: Transmission via BAM**

## 9.6 Transmission via CMDT

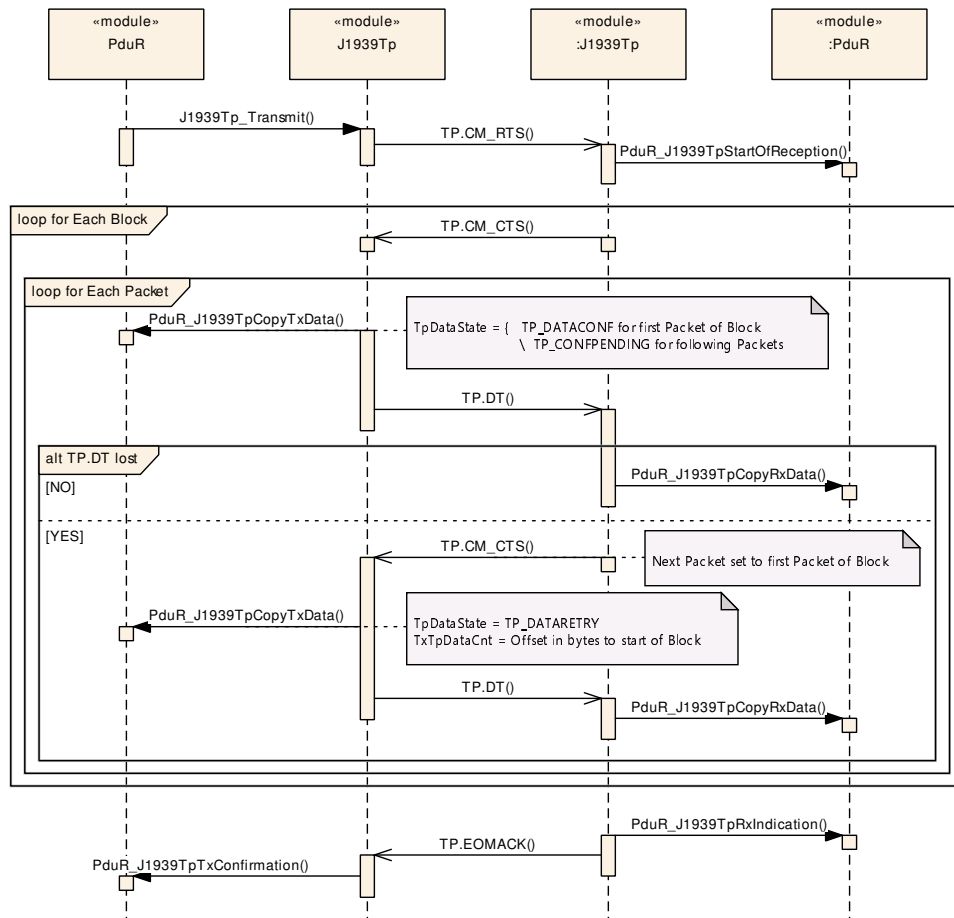
The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [PDU Router](#) and the [L-SDU Router](#) during transmission of a [PG](#) via [CMDT](#), i.e. an N-SDU that is larger than 8 bytes and is sent directly to the receiving node.



**Figure 9.6: Transmission via CMTD**

## 9.7 Handling of Retry during CMTD Transmission

The following diagram shows the interaction of the [SAE J1939 Transport Layer](#) module with the [PDU Router](#) in the sender and the receiver node during transmission of a [PG](#) via [CMTD](#) when a retry is performed because some data is lost.



**Figure 9.7: Retry Handling**

## 10 Configuration Specification

In general, this chapter defines configuration parameters and their clustering into containers. For general information about the definition of containers and parameters, refer to [16, SWS BSW General] 10.1 “Introduction to configuration specification”. For details about published information of the [SAE J1939 Transport Layer](#) module, refer to [16, SWS BSW General] 10.3 “Published Information”.

In [Chapter 10.1](#), the structure (containers) and the parameters of the module [J1939Tp](#) are specified.

### [SWS\_J1939Tp\_00084]

Upstream requirements: [SRS\\_BSW\\_00167](#)

[The consistency of the configuration must be checked by the configuration tool at configuration time. Configuration rules and constraints for plausibility checks will be performed where possible, during configuration time.]

### 10.1 Containers and Configuration Parameters

The following subsections summarize all configuration parameters. The detailed meanings of the parameters is described in chapters [7](#) and [8](#).

Some of these containers and parameters are derived from classes and attributes of the [20, TPS System Template], which also contains the rules for these derivations.

The following pictures show an overview of the configuration parameters available for [J1939Tp](#):

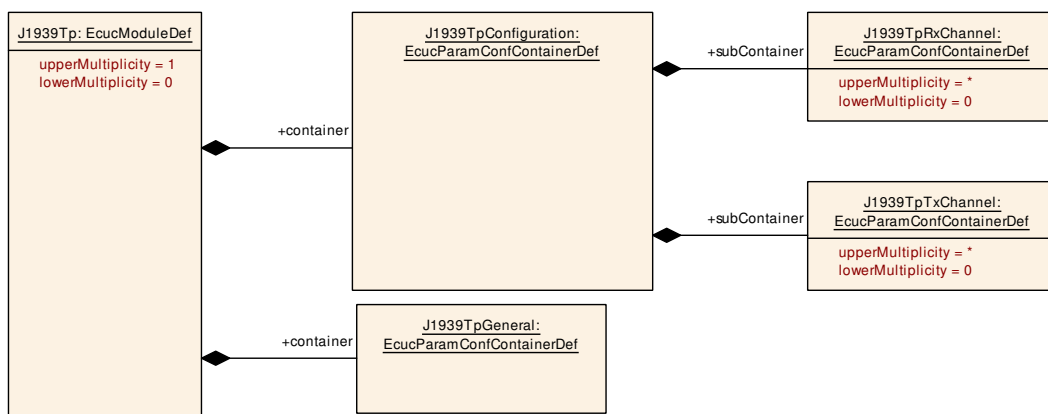


Figure 10.1: Module Configuration

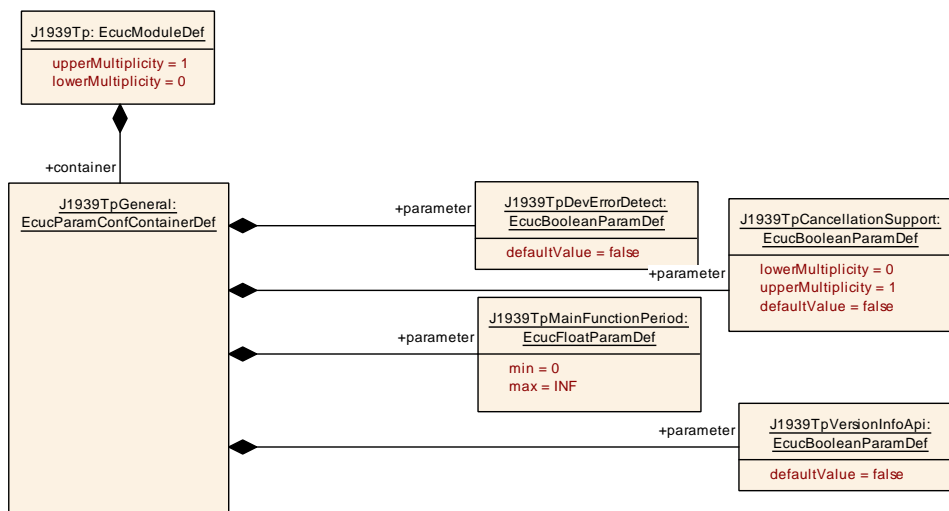


Figure 10.2: General Parameters

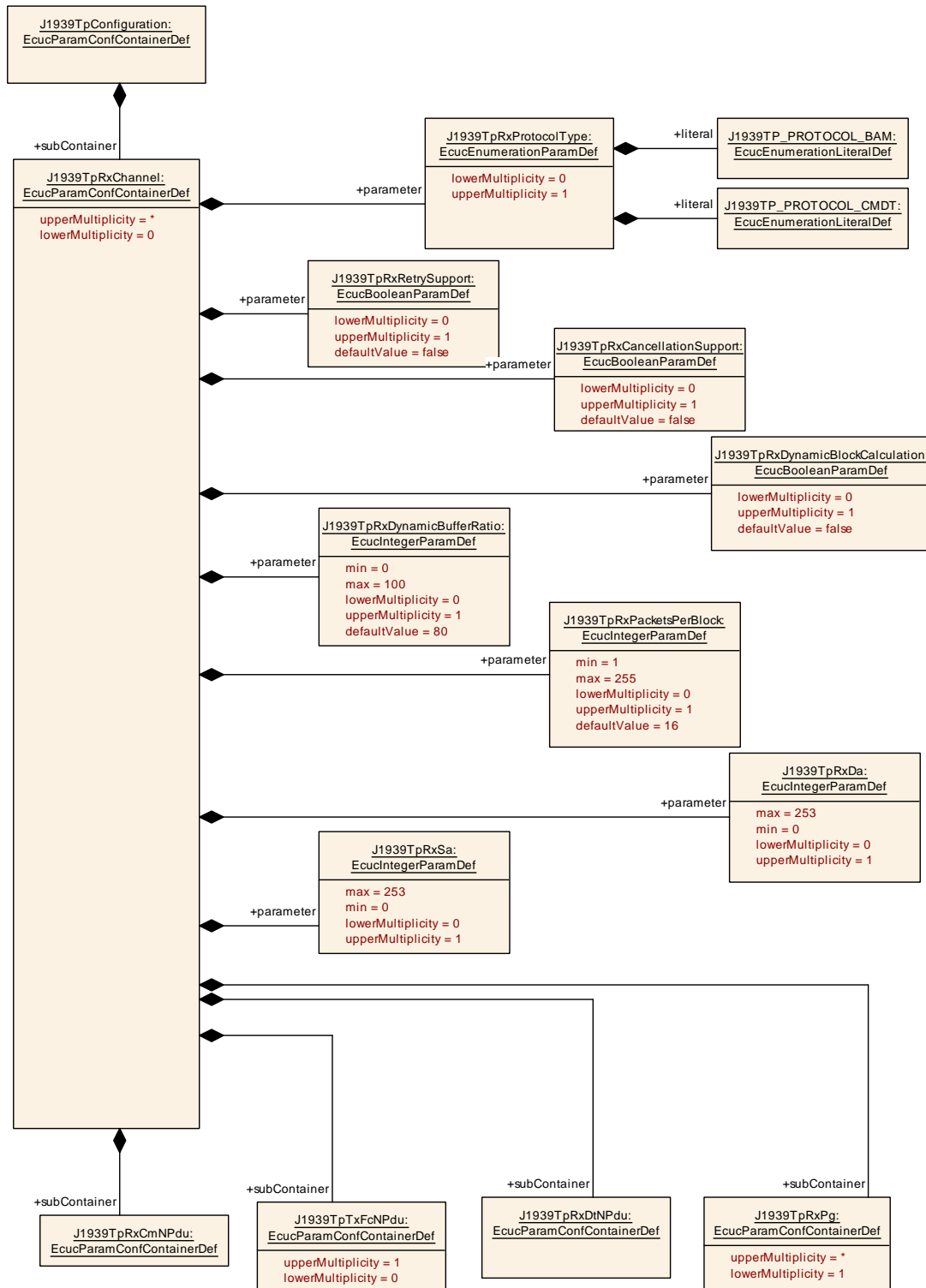
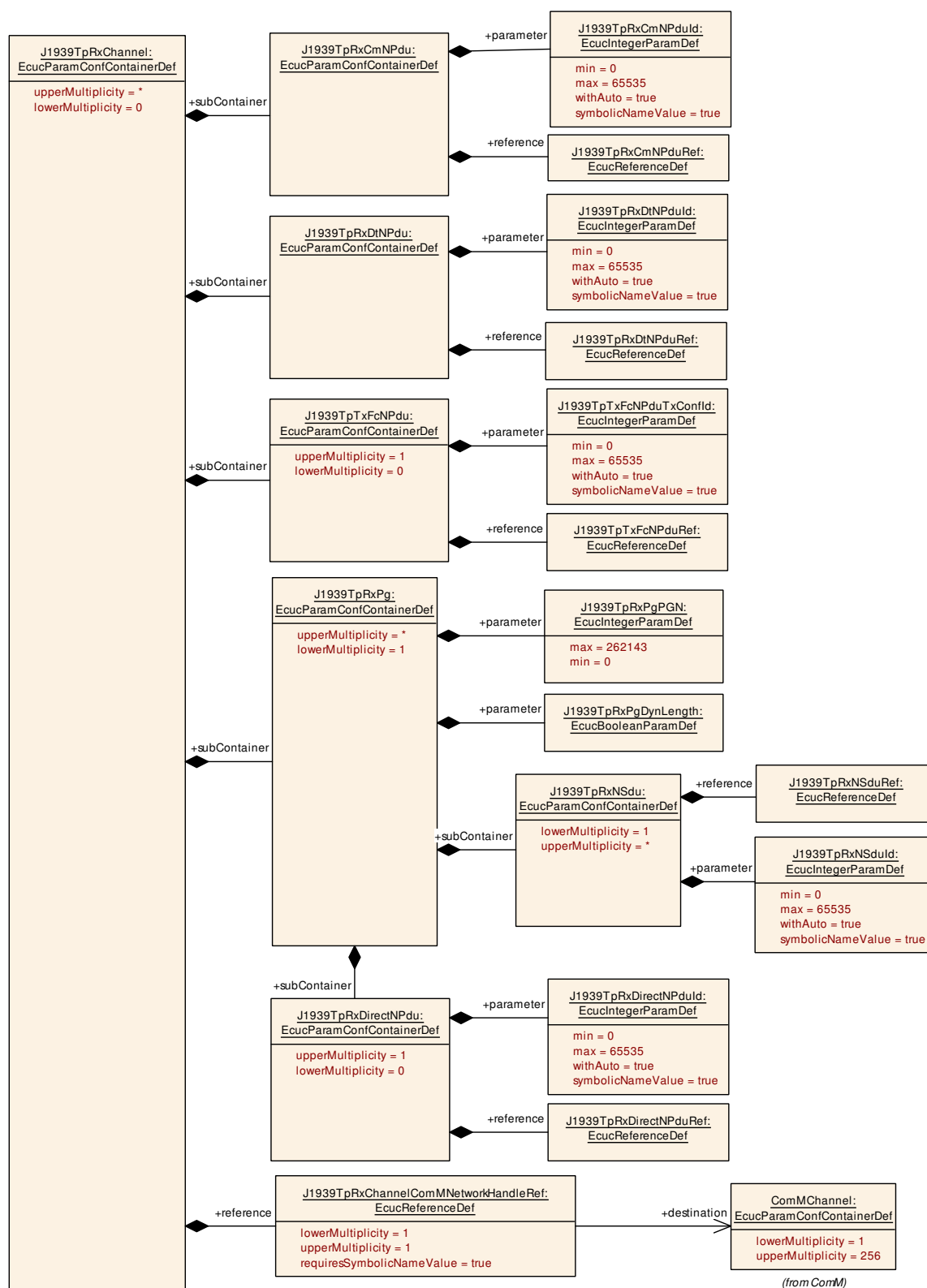


Figure 10.3: Configuration of Rx Channel - Part 1





### Figure 10.4: Configuration of Rx Channel - Part 2

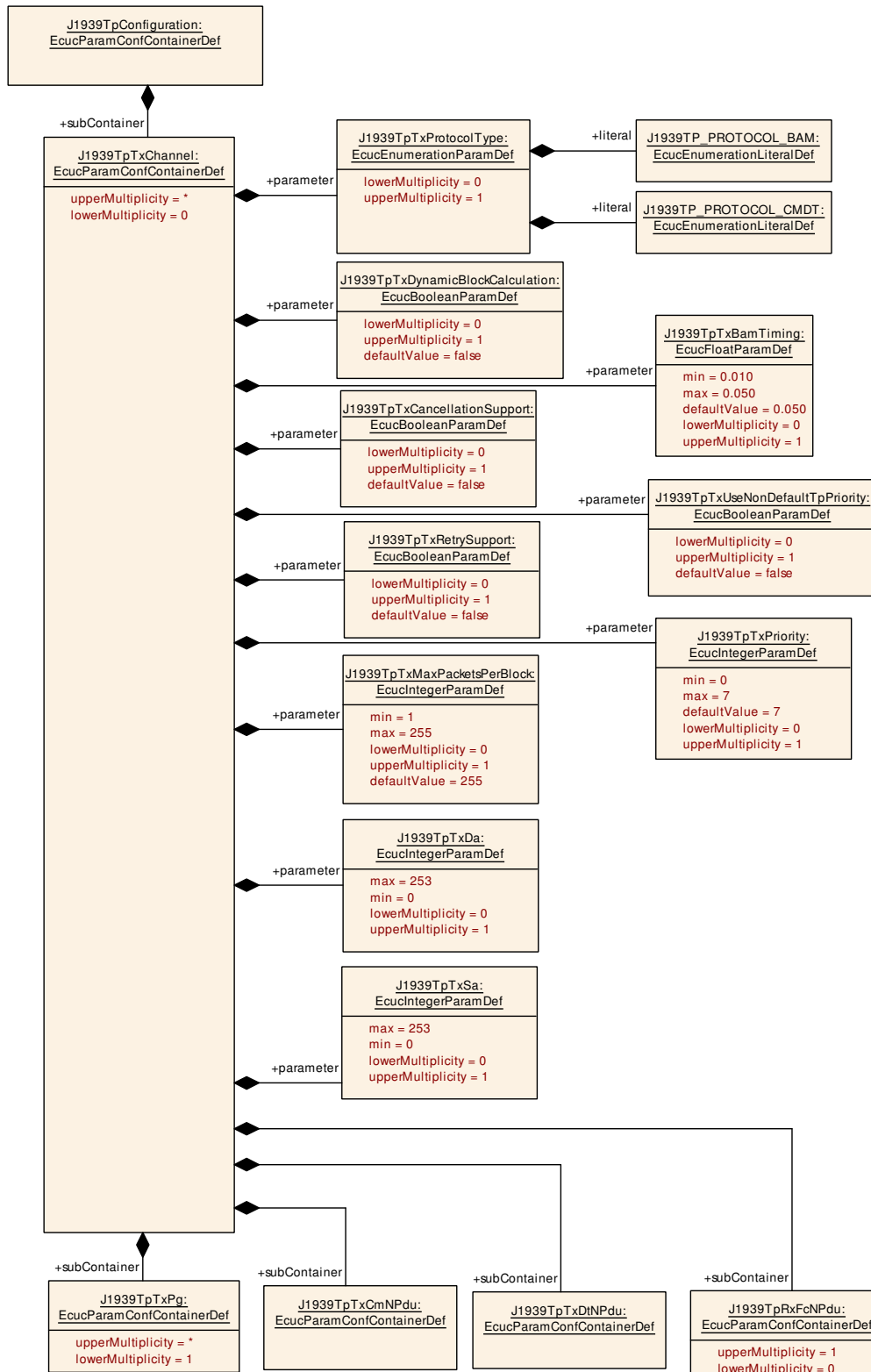


Figure 10.5: Configuration of Tx Channel - Part 1

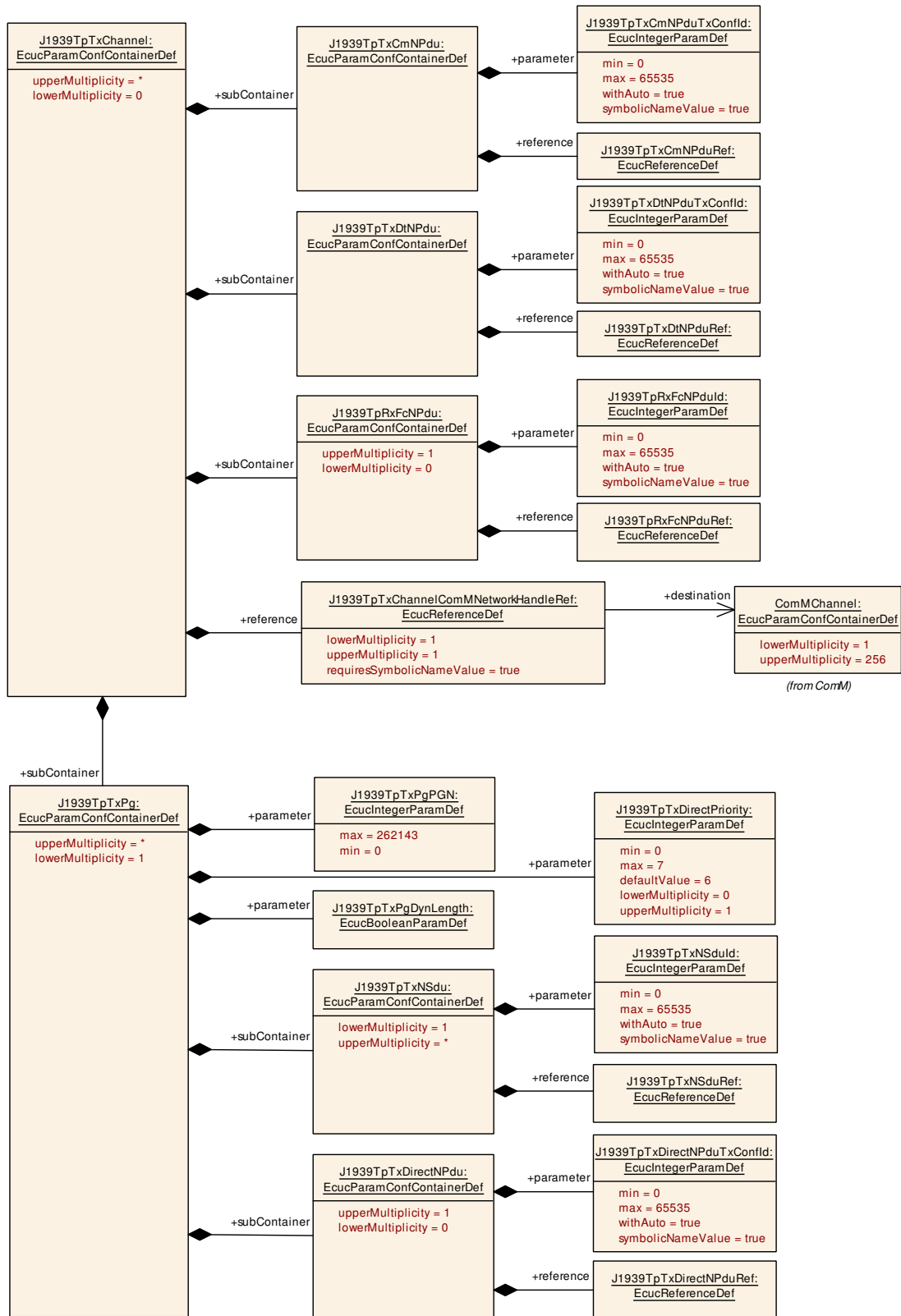


Figure 10.6: Configuration of Tx Channel - Part 2

As the [SAE J1939 Transport Layer](#) has interfaces to the [PDU Router](#), the configuration needs to reference the “global PDUs” (EcucPduCollection.Pdu) of the [21, TPS ECU Configuration].

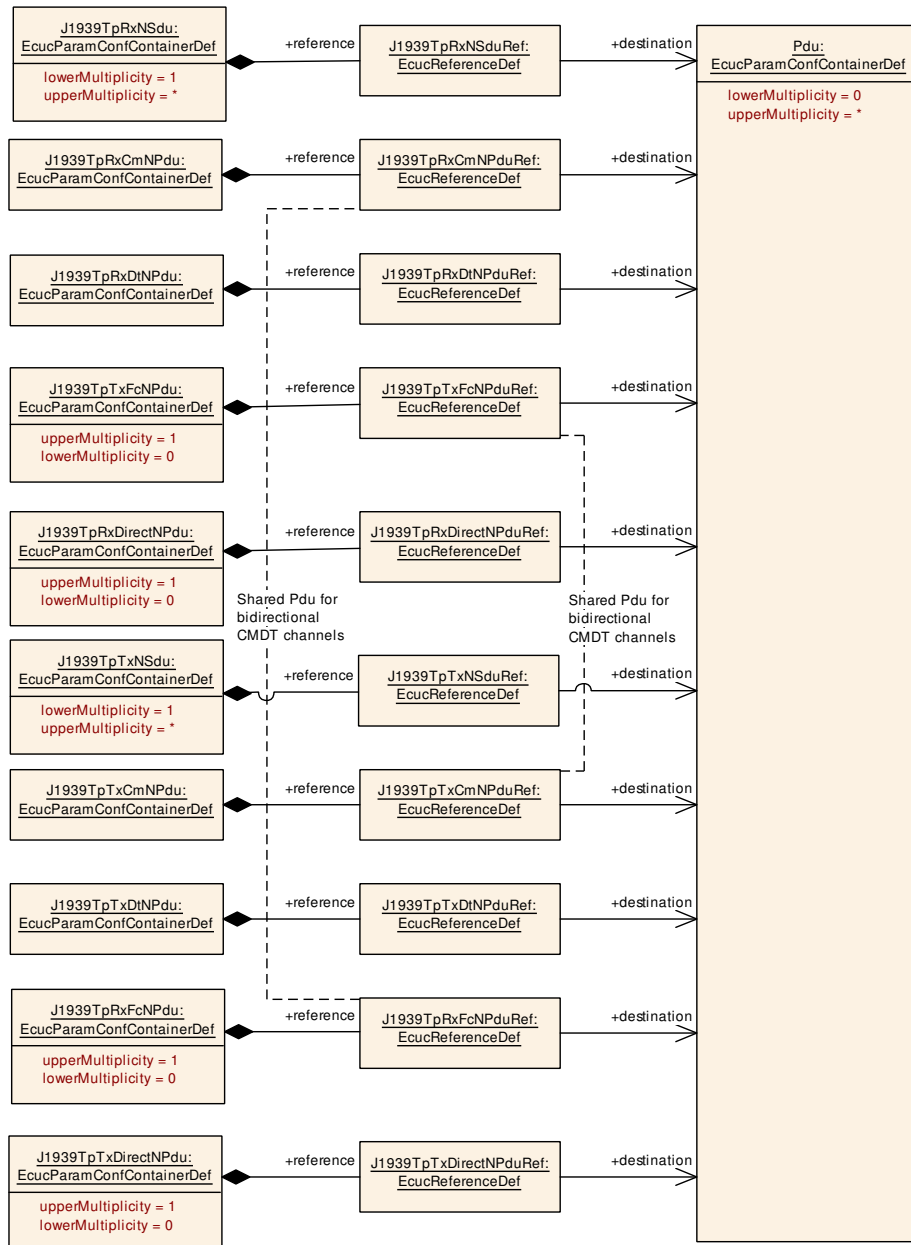


Figure 10.7: References to PDUs

### 10.1.1 J1939Tp

[ECUC\_J1939Tp\_00127] Definition of EcucModuleDef J1939Tp [

Module Name	J1939Tp
Description	Configuration of the J1939Tp (J1939 Transport Protocol) module.
Post-Build Variant Support	true
Supported Config Variants	VARIANT-LINK-TIME, VARIANT-POST-BUILD, VARIANT-PRE-COMPILE

Included Containers		
Container Name	Multiplicity	Dependency
<a href="#">J1939TpConfiguration</a>	1	This container contains the configuration parameters and sub containers of the J1939Tp module that define the communication paths.
<a href="#">J1939TpGeneral</a>	1	This container describes the general configuration parameters of the J1939Tp module.

### 10.1.2 J1939TpGeneral

#### [ECUC\_J1939Tp\_00033] Definition of EcucParamConfContainerDef J1939TpGeneral

Container Name	J1939TpGeneral
Parent Container	<a href="#">J1939Tp</a>
Description	This container describes the general configuration parameters of the J1939Tp module.
Multiplicity	1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpCancellationSupport</a>	0..1	[ECUC_J1939Tp_00174]
<a href="#">J1939TpDevErrorDetect</a>	1	[ECUC_J1939Tp_00042]
<a href="#">J1939TpMainFunctionPeriod</a>	1	[ECUC_J1939Tp_00044]
<a href="#">J1939TpVersionInfoApi</a>	1	[ECUC_J1939Tp_00051]

No Included Containers
------------------------

#### [ECUC\_J1939Tp\_00174] Definition of EcucBooleanParamDef J1939TpCancellationSupport

Parameter Name	J1939TpCancellationSupport
Parent Container	<a href="#">J1939TpGeneral</a>
Description	Enable transmit and receive cancellation. The APIs J1939Tp_CancelTransmit() and J1939Tp_CancelReceive() will only be available when this parameter is enabled.
Multiplicity	0..1
Type	EcucBooleanParamDef





Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00042] Definition of EcucBooleanParamDef J1939TpDevError Detect

Parameter Name	J1939TpDevErrorDetect		
Parent Container	<a href="#">J1939TpGeneral</a>		
Description	Switches the development error detection and notification on or off. <ul style="list-style-type: none"> <li>• true: detection and notification is enabled.</li> <li>• false: detection and notification is disabled.</li> </ul>		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00044] Definition of EcucFloatParamDef J1939TpMainFunction Period

Parameter Name	J1939TpMainFunctionPeriod		
Parent Container	<a href="#">J1939TpGeneral</a>		
Description	Allow to configure the time for the MainFunction (in seconds). Please note: This configuration value shall be equal to the value in the SchedulerManager module.		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00051] Definition of EcucBooleanParamDef J1939TpVersionInfoApi

Parameter Name	J1939TpVersionInfoApi		
Parent Container	<a href="#">J1939TpGeneral</a>		
Description	The function J1939Tp_GetVersionInfo is configurable (On/Off) by this configuration parameter.		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## 10.1.3 J1939TpConfiguration

### [ECUC\_J1939Tp\_00052] Definition of EcucParamConfContainerDef J1939TpConfiguration

Container Name	J1939TpConfiguration
Parent Container	<a href="#">J1939Tp</a>
Description	This container contains the configuration parameters and sub containers of the J1939 Tp module that define the communication paths.
Multiplicity	1
Configuration Parameters	

No Included Parameters
------------------------

Included Containers		
Container Name	Multiplicity	Dependency
<a href="#">J1939TpRxChannel</a>	0..*	This container describes a reception channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses.
<a href="#">J1939TpTxChannel</a>	0..*	This container describes a transmission channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses.

### 10.1.4 J1939TpRxChannel

#### [ECUC\_J1939Tp\_00053] Definition of EcucParamConfContainerDef J1939TpRxChannel

Container Name	J1939TpRxChannel
Parent Container	<a href="#">J1939TpConfiguration</a>
Description	This container describes a reception channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses.
Multiplicity	0..*
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxCancellationSupport</a>	0..1	<a href="#">[ECUC_J1939Tp_00186]</a>
<a href="#">J1939TpRxDa</a>	0..1	<a href="#">[ECUC_J1939Tp_00178]</a>
<a href="#">J1939TpRxDynamicBlockCalculation</a>	0..1	<a href="#">[ECUC_J1939Tp_00187]</a>
<a href="#">J1939TpRxDynamicBufferRatio</a>	0..1	<a href="#">[ECUC_J1939Tp_00188]</a>
<a href="#">J1939TpRxPacketsPerBlock</a>	0..1	<a href="#">[ECUC_J1939Tp_00189]</a>
<a href="#">J1939TpRxProtocolType</a>	0..1	<a href="#">[ECUC_J1939Tp_00029]</a>
<a href="#">J1939TpRxRetrySupport</a>	0..1	<a href="#">[ECUC_J1939Tp_00185]</a>
<a href="#">J1939TpRxSa</a>	0..1	<a href="#">[ECUC_J1939Tp_00179]</a>
<a href="#">J1939TpRxChannelComMNetworkHandleRef</a>	1	<a href="#">[ECUC_J1939Tp_00194]</a>

Included Containers		
Container Name	Multiplicity	Dependency
<a href="#">J1939TpRxCmNPdu</a>	1	This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. This N-PDU consumes a meta data item of type CAN_ID_32.
<a href="#">J1939TpRxDtNPdu</a>	1	This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. This N-PDU consumes a meta data item of type CAN_ID_32.
<a href="#">J1939TpRxPg</a>	1..*	Parameter group received by the J1939 transport layer.
<a href="#">J1939TpTxFcNPdu</a>	0..1	This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. This N-PDU produces a meta data item of type CAN_ID_32. Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all.



## [ECUC\_J1939Tp\_00186] Definition of EcucBooleanParamDef J1939TpRxCancellationSupport [

Parameter Name	J1939TpRxCancellationSupport		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Enable receive cancellation using the API J1939Tp_CancelReceive() for this channel.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00178] Definition of EcucIntegerParamDef J1939TpRxDa [

Parameter Name	J1939TpRxDa		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Destination address (DA) of this channel. This parameter is only required for channels with fixed DA which use N-PDUs with MetaData containing the DA.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 253		
Default value	–		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## [ECUC\_J1939Tp\_00187] Definition of EcucBooleanParamDef J1939TpRxDynamicBlockCalculation

Parameter Name	J1939TpRxDynamicBlockCalculation		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Enable dynamic calculation of "number of packets that can be sent" value in TPCM_CTS, based on the size of buffers in upper layers reported via StartOfReception and PduR_J1939TpCopyRxData.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00188] Definition of EcucIntegerParamDef J1939TpRxDynamicBufferRatio

Parameter Name	J1939TpRxDynamicBufferRatio		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Percentage of available buffer that shall be used for retry. This parameter is only applicable when "J1939TpRxRetrySupport" and "J1939TpRxDynamicBlockCalculation" are enabled.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 100		
Default value	80		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	Only applicable when "J1939TpRxRetrySupport" and "J1939TpRxDynamicBlockCalculation" are enabled		

## [ECUC\_J1939Tp\_00189] Definition of EcucIntegerParamDef J1939TpRxPacketsPerBlock

Parameter Name	J1939TpRxPacketsPerBlock		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Number of TP.DT frames the receiving J1939Tp module allows the sender to send before waiting for another TP.CM_CTS. This parameter is transmitted in the TP.CM_CTS frame, and is thus only relevant for reception of messages via CMDT. When J1939TpRxDynamicBlockCalculation is enabled, this parameter specifies a maximum for the calculated value. For further details on this parameter value see SAE J1939/21.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	1 .. 255		
Default value	16		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00029] Definition of EcucEnumerationParamDef J1939TpRxProtocolType

Parameter Name	J1939TpRxProtocolType		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Protocol type used by this channel. This parameter is only required for channels with fixed destination address.		
Multiplicity	0..1		
Type	EcucEnumerationParamDef		
Range	J1939TP_PROTOCOL_BAM	J1939 transport protocol type BAM (Broadcast Announce Message). This protocol uses two N-PDUs: The CmNPdu and the DtNPdu.	
	J1939TP_PROTOCOL_CMDT	J1939 transport protocol type CMDT (Connection Mode Data Transfer). This protocol uses three N-PDUs: The CmNPdu, the DtNPdu, and the FcNPdu.	
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00185] Definition of EcucBooleanParamDef J1939TpRxRetry Support

Parameter Name	J1939TpRxRetrySupport		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Enable support for triggering repetition of failed transmission using TPCM_CTS with a packet number that has already been sent. Retransmission is triggered when a sequence number is missing or a timeout occurs during reception.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00179] Definition of EcucIntegerParamDef J1939TpRxSa

Parameter Name	J1939TpRxSa		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Source address (SA) of this channel. This parameter is only required for channels with fixed SA which use N-PDUs with MetaData containing the SA.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 253		
Default value	–		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00194] Definition of EcucReferenceDef J1939TpRxChannel ComMNetworkHandleRef

Parameter Name	J1939TpRxChannelComMNetworkHandleRef		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Reference to the channel defined by the ComMChannel providing access to the unique channel index ComMChannelId.		
Multiplicity	1		
Type	Symbolic name reference to ComMChannel		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## 10.1.5 J1939TpRxCmNPdu

### [ECUC\_J1939Tp\_00128] Definition of EcucParamConfContainerDef J1939TpRx CmNPdu

Container Name	J1939TpRxCmNPdu		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMTD to initialize the connection. For CMTD, it is also used to abort the connection. This N-PDU consumes a meta data item of type CAN_ID_32.		
Multiplicity	1		
Configuration Parameters			

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxCmNPdulId</a>	1	[ECUC_J1939Tp_00129]
<a href="#">J1939TpRxCmNPduRef</a>	1	[ECUC_J1939Tp_00158]

No Included Containers
------------------------

### [ECUC\_J1939Tp\_00129] Definition of EcucIntegerParamDef J1939TpRxCmNPdu Id

Parameter Name	J1939TpRxCmNPdulId
Parent Container	<a href="#">J1939TpRxCmNPdu</a>
Description	The N-PDU identifier used for communication with LSduR.
Multiplicity	1
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)



△

Range	0 .. 65535	
Default value	–	
Post-Build Variant Value	false	
Value Configuration Class	Pre-compile time	X All Variants
	Link time	–
	Post-build time	–
Dependency	withAuto = true	

]

## [ECUC\_J1939Tp\_00158] Definition of EcucReferenceDef J1939TpRxCmNPduRef

[

Parameter Name	J1939TpRxCmNPduRef		
Parent Container	<a href="#">J1939TpRxCmNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

]

### 10.1.6 J1939TpRxDtNPdu

## [ECUC\_J1939Tp\_00117] Definition of EcucParamConfContainerDef J1939TpRxDtNPdu

[

Container Name	J1939TpRxDtNPdu
Parent Container	<a href="#">J1939TpRxChannel</a>
Description	This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CDMT to transfer the contents of an N-SDU. This N-PDU consumes a meta data item of type CAN_ID_32.
Multiplicity	1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxDtNPduId</a>	1	[ECUC_J1939Tp_00133]
<a href="#">J1939TpRxDtNPduRef</a>	1	[ECUC_J1939Tp_00134]

No Included Containers
------------------------

]

### [ECUC\_J1939Tp\_00133] Definition of EcucIntegerParamDef J1939TpRxDtNPduId

Parameter Name	J1939TpRxDtNPduId		
Parent Container	<a href="#">J1939TpRxDtNPdu</a>		
Description	The N-PDU identifier used for communication with LSduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

### [ECUC\_J1939Tp\_00134] Definition of EcucReferenceDef J1939TpRxDtNPduRef

Parameter Name	J1939TpRxDtNPduRef		
Parent Container	<a href="#">J1939TpRxDtNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## 10.1.7 J1939TpRXPg

### [ECUC\_J1939Tp\_00050] Definition of EcucParamConfContainerDef J1939TpRXPg

Container Name	J1939TpRxPg		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	Parameter group received by the J1939 transport layer.		
Multiplicity	1..*		
Configuration Parameters			
Included Parameters			
Parameter Name	Multiplicity	ECUC ID	
<a href="#">J1939TpRxPgDynLength</a>	1	<a href="#">[ECUC_J1939Tp_00066]</a>	
<a href="#">J1939TpRxPgPGN</a>	1	<a href="#">[ECUC_J1939Tp_00065]</a>	

Included Containers		
Container Name	Multiplicity	Dependency
<a href="#">J1939TpRxDirectNPdu</a>	0..1	This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less than 8 bytes. This N-PDU consumes a meta data item of type CAN_ID_32. Please note: This sub container is only necessary when J1939TpRxPgDynLength is TRUE.
<a href="#">J1939TpRxNSdu</a>	1..*	This container describes the parameters that are relevant for the reception of a specific N-SDU. This N-SDU produces meta data items of type SOURCE_ADDRESS_16, TARGET_ADDRESS_16, and PRIORITY_8.

### [ECUC\_J1939Tp\_00066] Definition of EcucBooleanParamDef J1939TpRxPgDynLength

Parameter Name	J1939TpRxPgDynLength		
Parent Container	<a href="#">J1939TpRxPg</a>		
Description	This flag is set to TRUE when the N-SDU refers to a PGN with variable length. Please note: When this attribute is TRUE, the sub container J1939TpRxDirectNPdu is required.		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00065] Definition of EcucIntegerParamDef J1939TpRxPgPGN

Parameter Name	J1939TpRxPgPGN		
Parent Container	<a href="#">J1939TpRxPg</a>		
Description	PGN of the referenced N-SDUs.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 262143		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			



### 10.1.8 J1939TpRxDirectNPdu

#### [ECUC\_J1939Tp\_00130] Definition of EcucParamConfContainerDef J1939TpRxDirectNPdu

Container Name	J1939TpRxDirectNPdu
Parent Container	<a href="#">J1939TpRxPg</a>
Description	This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less than 8 bytes. This N-PDU consumes a meta data item of type CAN_ID_32. Please note: This sub container is only necessary when J1939TpRxPgDynLength is TRUE.
Multiplicity	0..1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxDirectNPduld</a>	1	[ECUC_J1939Tp_00131]
<a href="#">J1939TpRxDirectNPduRef</a>	1	[ECUC_J1939Tp_00132]

No Included Containers

#### [ECUC\_J1939Tp\_00131] Definition of EcucIntegerParamDef J1939TpRxDirectNPduld

Parameter Name	J1939TpRxDirectNPduld		
Parent Container	<a href="#">J1939TpRxDirectNPdu</a>		
Description	The N-PDU identifier used for communication with LSduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

#### [ECUC\_J1939Tp\_00132] Definition of EcucReferenceDef J1939TpRxDirectNPduRef

Parameter Name	J1939TpRxDirectNPduRef
Parent Container	<a href="#">J1939TpRxDirectNPdu</a>
Description	Reference to the Pdu object representing the N-PDU.
Multiplicity	1
Type	Reference to Pdu





Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

]

### 10.1.9 J1939TpRxNSdu

#### [ECUC\_J1939Tp\_00063] Definition of EcucParamConfContainerDef J1939TpRxNSdu

Container Name	J1939TpRxNSdu
Parent Container	<a href="#">J1939TpRxPg</a>
Description	This container describes the parameters that are relevant for the reception of a specific N-SDU. This N-SDU produces meta data items of type SOURCE_ADDRESS_16, TARGET_ADDRESS_16, and PRIORITY_8.
Multiplicity	1..*
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxNSduld</a>	1	<a href="#">[ECUC_J1939Tp_00184]</a>
<a href="#">J1939TpRxNSduRef</a>	1	<a href="#">[ECUC_J1939Tp_00069]</a>

No Included Containers
------------------------

]

#### [ECUC\_J1939Tp\_00184] Definition of EcucIntegerParamDef J1939TpRxNSduld

Parameter Name	J1939TpRxNSduld		
Parent Container	<a href="#">J1939TpRxNSdu</a>		
Description	This is a unique identifier for a received N-SDU. This Id is used in the CancelReceive API call.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

]

### [ECUC\_J1939Tp\_00069] Definition of EcucReferenceDef J1939TpRxNSduRef [

Parameter Name	J1939TpRxNSduRef		
Parent Container	<a href="#">J1939TpRxNSdu</a>		
Description	Reference to the Pdu object representing the N-SDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### 10.1.10 J1939TpTxFcNPdu

### [ECUC\_J1939Tp\_00135] Definition of EcucParamConfContainerDef J1939TpTx FcNPdu [

Container Name	J1939TpTxFcNPdu		
Parent Container	<a href="#">J1939TpRxChannel</a>		
Description	This N-PDU represents the TPCM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TPCM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. This N-PDU produces a meta data item of type CAN_ID_32. Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all.		
Multiplicity	0..1		
Configuration Parameters			

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpTxFcNPduTxConfId</a>	1	[ECUC_J1939Tp_00168]
<a href="#">J1939TpTxFcNPduRef</a>	1	[ECUC_J1939Tp_00136]

No Included Containers
------------------------

### [ECUC\_J1939Tp\_00168] Definition of EcucIntegerParamDef J1939TpTxFcNPdu TxConfId [

Parameter Name	J1939TpTxFcNPduTxConfId		
Parent Container	<a href="#">J1939TpTxFcNPdu</a>		
Description	The N-PDU identifier used for Tx confirmation from LSduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		





Range	0 .. 65535	
Default value	–	
Post-Build Variant Value	false	
Value Configuration Class	Pre-compile time	X All Variants
	Link time	–
	Post-build time	–
Dependency	withAuto = true	

### [ECUC\_J1939Tp\_00136] Definition of EcucReferenceDef J1939TpTxFcNPduRef

Parameter Name	J1939TpTxFcNPduRef		
Parent Container	<a href="#">J1939TpTxFcNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU. Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpTxCmNPdu Ref of the corresponding J1939TpTxChannel.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## 10.1.11 J1939TpTxChannel

### [ECUC\_J1939Tp\_00059] Definition of EcucParamConfContainerDef J1939TpTxChannel

Container Name	J1939TpTxChannel
Parent Container	<a href="#">J1939TpConfiguration</a>
Description	This container describes a transmission channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses.
Multiplicity	0..*
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
J1939TpTxBamTiming	0..1	[ECUC_J1939Tp_00197]
J1939TpTxCancellationSupport	0..1	[ECUC_J1939Tp_00192]
J1939TpTxDa	0..1	[ECUC_J1939Tp_00180]
J1939TpTxDynamicBlockCalculation	0..1	[ECUC_J1939Tp_00191]
J1939TpTxMaxPacketsPerBlock	0..1	[ECUC_J1939Tp_00190]
J1939TpTxPriority	0..1	[ECUC_J1939Tp_00199]
J1939TpTxProtocolType	0..1	[ECUC_J1939Tp_00137]
J1939TpTxRetrySupport	0..1	[ECUC_J1939Tp_00193]
J1939TpTxSa	0..1	[ECUC_J1939Tp_00181]
J1939TpTxUseNonDefaultTpPriority	0..1	[ECUC_J1939Tp_00198]
J1939TpTxChannelComMNetworkHandleRef	1	[ECUC_J1939Tp_00195]

Included Containers		
Container Name	Multiplicity	Dependency
J1939TpRxFcNPdu	0..1	This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. This N-PDU consumes a meta data item of type CAN_ID_32. Please note: This sub container is only required when J1939TpTxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all.
J1939TpTxCmNPdu	1	This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. This N-PDU produces a meta data item of type CAN_ID_32.
J1939TpTxDtNPdu	1	This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. This N-PDU produces a meta data item of type CAN_ID_32.
J1939TpTxPg	1..*	Parameter group transmitted by the J1939 transport layer.

## [ECUC\_J1939Tp\_00197] Definition of EcucFloatParamDef J1939TpTxBamTiming

Parameter Name	J1939TpTxBamTiming		
Parent Container	J1939TpTxChannel		
Description	If configured, TP.DT frames of a BAM transmission will be sent with the configured delay in seconds. If not configured, 0,05 seconds will be used.		
Multiplicity	0..1		
Type	EcucFloatParamDef		
Range	[0.01 .. 0.05]		
Default value	0.05		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE





Value Configuration Class	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00192] Definition of EcucBooleanParamDef J1939TpTxCancellationSupport [

Parameter Name	J1939TpTxCancellationSupport		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Enable transmit cancellation using the API J1939Tp_CancelTransmit() for this channel.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00180] Definition of EcucIntegerParamDef J1939TpTxDa [

Parameter Name	J1939TpTxDa		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Destination address (DA) of this channel. This parameter is only required for channels with fixed DA which use N-PDUs with MetaData containing the DA.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 253		
Default value	—		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD





Dependency	
------------	--

### [ECUC\_J1939Tp\_00191] Definition of EcucBooleanParamDef J1939TpTxDynamicBlockCalculation

Parameter Name	J1939TpTxDynamicBlockCalculation		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Enable dynamic calculation of "maximum number of packets that can be sent" value in TP.CM_RTS, based on the available amount of data in upper layers reported via PduR_J1939TpCopyTxData.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00190] Definition of EcucIntegerParamDef J1939TpTxMaxPacketsPerBlock

Parameter Name	J1939TpTxMaxPacketsPerBlock		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Maximum number of TP.DT frames the transmitting J1939Tp module is ready to send before waiting for another TP.CM_CTS. This parameter is transmitted in the TP.CM_RTS frame, and is thus only relevant for transmission of messages via CMDT. When J1939TpTxDynamicBlockCalculation is enabled, this parameter specifies a maximum for the calculated value. For further details on this parameter value see SAE J1939/21.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	1 .. 255		
Default value	255		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	





Dependency	
------------	--

### [ECUC\_J1939Tp\_00199] Definition of EcucIntegerParamDef J1939TpTxPriority [

Parameter Name	J1939TpTxPriority		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	If configured, and J1939TpTxUseNonDefaultTpPriority is enabled, this value will be used as priority of transmitted TP.CM and TP.DT frames, otherwise the default priority 7 will be used.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 7		
Default value	7		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00137] Definition of EcucEnumerationParamDef J1939TpTxProtocolType [

Parameter Name	J1939TpTxProtocolType		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Protocol type used by this channel. This parameter is only required for channels with fixed destination address.		
Multiplicity	0..1		
Type	EcucEnumerationParamDef		
Range	J1939TP_PROTOCOL_BAM	J1939 transport protocol type BAM (Broadcast Announce Message). This protocol uses two N-PDUs: The CmNPdu and the DtNPdu.	
	J1939TP_PROTOCOL_CMDT	J1939 transport protocol type CMDT (Connection Mode Data Transfer). This protocol uses three N-PDUs: The CmNPdu, the DtNPdu, and the FcNPdu.	
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	







	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00193] Definition of EcucBooleanParamDef J1939TpTxRetry Support

Parameter Name	J1939TpTxRetrySupport		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Enable support for repetition of failed transmission using TP.CM_CTS with a packet number that has already been sent. Retransmission is handled via the retry feature of PduR_J1939TpCopyTxData.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## [ECUC\_J1939Tp\_00181] Definition of EcucIntegerParamDef J1939TpTxSa

Parameter Name	J1939TpTxSa		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Source address (SA) of this channel. This parameter is only required for channels with fixed SA which use N-PDUs with MetaData containing the SA.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 253		
Default value	—		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00198] Definition of EcucBooleanParamDef J1939TpTxUseNonDefaultTpPriority [

Parameter Name	J1939TpTxUseNonDefaultTpPriority		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Enables/disables non-default priority of transmitted TP.CM and TP.DT frames. If enabled, the priority will be set according to J1939TpTxPriority or to the priority provided by the upper layer via meta data.		
Multiplicity	0..1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00195] Definition of EcucReferenceDef J1939TpTxChannelComMNetworkHandleRef [

Parameter Name	J1939TpTxChannelComMNetworkHandleRef		
Parent Container	<a href="#">J1939TpTxChannel</a>		
Description	Reference to the channel defined by the ComMChannel providing access to the unique channel index ComMChannelId.		
Multiplicity	1		
Type	Symbolic name reference to ComMChannel		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

## 10.1.12 J1939TpRxFcNPdu

### [ECUC\_J1939Tp\_00144] Definition of EcucParamConfContainerDef J1939TpRx FcNPdu [

<b>Container Name</b>	J1939TpRxFcNPdu
<b>Parent Container</b>	<a href="#">J1939TpTxChannel</a>
<b>Description</b>	This N-PDU represents the TPCM frame that is used in reverse direction for a J1939 transport protocol session using the CMTD protocol type. TPCM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. This N-PDU consumes a meta data item of type CAN_ID_32. Please note: This sub container is only required when J1939TpTxProtocolType is J1939TP_PROTOCOL_CMTD or when it is not configured at all.
<b>Multiplicity</b>	0..1
<b>Configuration Parameters</b>	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpRxFcNPduld</a>	1	[ECUC_J1939Tp_00145]
<a href="#">J1939TpRxFcNPduRef</a>	1	[ECUC_J1939Tp_00146]

No Included Containers
------------------------

]

## [ECUC\_J1939Tp\_00145] Definition of EcucIntegerParamDef J1939TpRxFcNPduId

<b>Parameter Name</b>	J1939TpRxFcNPduld		
<b>Parent Container</b>	<a href="#">J1939TpRxFcNPdu</a>		
<b>Description</b>	The N-PDU identifier used for communication with LSduR.		
<b>Multiplicity</b>	1		
<b>Type</b>	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
<b>Range</b>	0 .. 65535		
<b>Default value</b>	–		
<b>Post-Build Variant Value</b>	false		
<b>Value Configuration Class</b>	<b>Pre-compile time</b>	X	All Variants
	<b>Link time</b>	–	
	<b>Post-build time</b>	–	
<b>Dependency</b>	withAuto = true		

]

## [ECUC\_J1939Tp\_00146] Definition of EcucReferenceDef J1939TpRxFcNPduRef

<b>Parameter Name</b>	J1939TpRxFcNPduRef		
<b>Parent Container</b>	<a href="#">J1939TpRxFcNPdu</a>		
<b>Description</b>	Reference to the Pdu object representing the N-PDU. Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpRxCmNPduRef of the corresponding J1939TpRxChannel.		
<b>Multiplicity</b>	1		
<b>Type</b>	Reference to Pdu		
<b>Post-Build Variant Value</b>	true		





Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

]

### 10.1.13 J1939TpTxCmNPdu

#### [ECUC\_J1939Tp\_00138] Definition of EcucParamConfContainerDef J1939TpTxCmNPdu [

Container Name	J1939TpTxCmNPdu
Parent Container	<a href="#">J1939TpTxChannel</a>
Description	This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. This N-PDU produces a meta data item of type CAN_ID_32.
Multiplicity	1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpTxCmNPduTxConfId</a>	1	<a href="#">[ECUC_J1939Tp_00170]</a>
<a href="#">J1939TpTxCmNPduRef</a>	1	<a href="#">[ECUC_J1939Tp_00139]</a>

No Included Containers
------------------------

]

#### [ECUC\_J1939Tp\_00170] Definition of EcucIntegerParamDef J1939TpTxCmNPduTxConfId [

Parameter Name	J1939TpTxCmNPduTxConfId		
Parent Container	<a href="#">J1939TpTxCmNPdu</a>		
Description	The N-PDU identifier used for Tx confirmation from LSduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

]

## [ECUC\_J1939Tp\_00139] Definition of EcucReferenceDef J1939TpTxCmNPduRef

Parameter Name	J1939TpTxCmNPduRef		
Parent Container	<a href="#">J1939TpTxCmNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### 10.1.14 J1939TpTxDtNPdu

## [ECUC\_J1939Tp\_00142] Definition of EcucParamConfContainerDef J1939TpTxDtNPdu

Container Name	J1939TpTxDtNPdu
Parent Container	<a href="#">J1939TpTxChannel</a>
Description	This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. This N-PDU produces a meta data item of type CAN_ID_32.
Multiplicity	1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpTxDtNPduTxConflD</a>	1	[ECUC_J1939Tp_00171]
<a href="#">J1939TpTxDtNPduRef</a>	1	[ECUC_J1939Tp_00143]

No Included Containers
------------------------

## [ECUC\_J1939Tp\_00171] Definition of EcucIntegerParamDef J1939TpTxDtNPduTxConflD

Parameter Name	J1939TpTxDtNPduTxConflD	
Parent Container	<a href="#">J1939TpTxDtNPdu</a>	
Description	The N-PDU identifier used for Tx confirmation from LSduR.	
Multiplicity	1	
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)	
Range	0 .. 65535	





Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

### [ECUC\_J1939Tp\_00143] Definition of EcucReferenceDef J1939TpTxDtNPduRef [

Parameter Name	J1939TpTxDtNPduRef		
Parent Container	<a href="#">J1939TpTxDtNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## 10.1.15 J1939TpTxPg

### [ECUC\_J1939Tp\_00070] Definition of EcucParamConfContainerDef J1939TpTxPg [

Container Name	J1939TpTxPg
Parent Container	<a href="#">J1939TpTxChannel</a>
Description	Parameter group transmitted by the J1939 transport layer.
Multiplicity	1..*
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpTxDirectPriority</a>	0..1	[ECUC_J1939Tp_00200]
<a href="#">J1939TpTxPgDynLength</a>	1	[ECUC_J1939Tp_00148]
<a href="#">J1939TpTxPgPgn</a>	1	[ECUC_J1939Tp_00150]

Included Containers		
Container Name	Multiplicity	Dependency
J1939TpTxDirectNPdu	0..1	This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less than 8 bytes. This N-PDU produces a meta data item of type CAN_ID_32. Please note: This sub container is only necessary when J1939TpTxPgDynLength is TRUE.
J1939TpTxNSdu	1..*	This container describes the parameters that are relevant for the transmission of a specific N-SDU. This N-SDU consumes meta data items of type SOURCE_ADDRESS_16, TARGET_ADDRESS_16, and PRIORITY_8.

### [ECUC\_J1939Tp\_00200] Definition of EcucIntegerParamDef J1939TpTxDirectPriority

Parameter Name	J1939TpTxDirectPriority		
Parent Container	J1939TpTxPg		
Description	If configured, and J1939TpTxUseNonDefaultTpPriority is enabled for the enclosing J1939TpTxChannel, this value will be used as priority of transmitted direct frames, otherwise the default priority 6 will be used.		
Multiplicity	0..1		
Type	EcucIntegerParamDef		
Range	0 .. 7		
Default value	6		
Post-Build Variant Multiplicity	true		
Post-Build Variant Value	true		
Multiplicity Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

### [ECUC\_J1939Tp\_00148] Definition of EcucBooleanParamDef J1939TpTxPgDynLength

Parameter Name	J1939TpTxPgDynLength		
Parent Container	J1939TpTxPg		
Description	This flag is set to TRUE when the N-SDU refers to a PGN with variable length. Please note: When this attribute is TRUE, the sub container J1939TpTxDirectNPdu is required.		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	





	Post-build time	–	
Dependency			

### [ECUC\_J1939Tp\_00150] Definition of EcucIntegerParamDef J1939TpTxPgPGN

Parameter Name	J1939TpTxPgPGN		
Parent Container	<a href="#">J1939TpTxPg</a>		
Description	PGN of the referenced N-SDUs.		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 262143		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency			

### 10.1.16 J1939TpTxDirectNPdu

### [ECUC\_J1939Tp\_00140] Definition of EcucParamConfContainerDef J1939TpTxDirectNPdu

Container Name	J1939TpTxDirectNPdu
Parent Container	<a href="#">J1939TpTxPg</a>
Description	This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less than 8 bytes. This N-PDU produces a meta data item of type CAN_ID_32. Please note: This sub container is only necessary when J1939TpTxPgDynLength is TRUE.
Multiplicity	0..1
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
<a href="#">J1939TpTxDirectNPduTxConfId</a>	1	<a href="#">[ECUC_J1939Tp_00169]</a>
<a href="#">J1939TpTxDirectNPduRef</a>	1	<a href="#">[ECUC_J1939Tp_00141]</a>

No Included Containers
------------------------



### [ECUC\_J1939Tp\_00169] Definition of EcucIntegerParamDef J1939TpTxDirectNPduTxConfId

Parameter Name	J1939TpTxDirectNPduTxConfId		
Parent Container	<a href="#">J1939TpTxDirectNPdu</a>		
Description	The N-PDU identifier used for Tx confirmation from LSduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

### [ECUC\_J1939Tp\_00141] Definition of EcucReferenceDef J1939TpTxDirectNPduRef

Parameter Name	J1939TpTxDirectNPduRef		
Parent Container	<a href="#">J1939TpTxDirectNPdu</a>		
Description	Reference to the Pdu object representing the N-PDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## 10.1.17 J1939TpTxNSdu

### [ECUC\_J1939Tp\_00147] Definition of EcucParamConfContainerDef J1939TpTxNSdu

Container Name	J1939TpTxNSdu
Parent Container	<a href="#">J1939TpTxPg</a>
Description	This container describes the parameters that are relevant for the transmission of a specific N-SDU. This N-SDU consumes meta data items of type SOURCE_ADDRESS_16, TARGET_ADDRESS_16, and PRIORITY_8.
Multiplicity	1..*
Configuration Parameters	

Included Parameters		
Parameter Name	Multiplicity	ECUC ID
J1939TpTxNSduld	1	[ECUC_J1939Tp_00149]
J1939TpTxNSduRef	1	[ECUC_J1939Tp_00151]

No Included Containers
------------------------

### [ECUC\_J1939Tp\_00149] Definition of EcucIntegerParamDef J1939TpTxNSduld [

Parameter Name	J1939TpTxNSduld		
Parent Container	J1939TpTxNSdu		
Description	The N-SDU identifier used for communication with PduR.		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Dependency	withAuto = true		

### [ECUC\_J1939Tp\_00151] Definition of EcucReferenceDef J1939TpTxNSduRef [

Parameter Name	J1939TpTxNSduRef		
Parent Container	J1939TpTxNSdu		
Description	Reference to the Pdu object representing the N-SDU.		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	true		
Value Configuration Class	Pre-compile time	X	VARIANT-PRE-COMPILE
	Link time	X	VARIANT-LINK-TIME
	Post-build time	X	VARIANT-POST-BUILD
Dependency			

## A Not Applicable Requirements

### [SWS\_J1939Tp\_NA\_00001] Requirements Not Applicable to this Specification

*Upstream requirements:* SRS\_BSW\_00168, SRS\_BSW\_00170, SRS\_BSW\_00375, SRS\_BSW\_00383, SRS\_BSW\_00384, SRS\_BSW\_00388, SRS\_BSW\_00389, SRS\_BSW\_00390, SRS\_BSW\_00392, SRS\_BSW\_00393, SRS\_BSW\_00395, SRS\_BSW\_00403, SRS\_BSW\_00416, SRS\_BSW\_00417, SRS\_BSW\_00419, SRS\_BSW\_00422, SRS\_BSW\_00423, SRS\_BSW\_00425, SRS\_BSW\_00427, SRS\_BSW\_00433, SRS\_BSW\_00437, SRS\_BSW\_00469, SRS\_BSW\_00471, SRS\_BSW\_00472, SRS\_BSW\_00490, SRS\_BSW\_00491

[These requirements are not applicable to this specification.]

### [SWS\_J1939Tp\_NA\_00002] Only One Main Function

*Upstream requirements:* SRS\_BSW\_00432

[The [SAE J1939 Transport Layer](#) usually does not have much to do in its main function, therefore a single main function is sufficient.]

## B Change History of AUTOSAR Traceable Items

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

### B.1 Traceable Item History of this Document According to AUTOSAR Release R25-11

#### B.1.1 Added Specification Items in R25-11

[\[ECUC\\_J1939Tp\\_00197\]](#) [\[ECUC\\_J1939Tp\\_00198\]](#) [\[ECUC\\_J1939Tp\\_00199\]](#)  
[\[ECUC\\_J1939Tp\\_00200\]](#) [\[SWS\\_J1939Tp\\_00236\]](#) [\[SWS\\_J1939Tp\\_00237\]](#) [\[SWS\\_J1939Tp\\_00238\]](#)

#### B.1.2 Changed Specification Items in R25-11

[\[ECUC\\_J1939Tp\\_00184\]](#) [\[SWS\\_J1939Tp\\_00041\]](#) [\[SWS\\_J1939Tp\\_00045\]](#) [\[SWS\\_J1939Tp\\_00121\]](#) [\[SWS\\_J1939Tp\\_00162\]](#) [\[SWS\\_J1939Tp\\_00230\]](#)

#### B.1.3 Deleted Specification Items in R25-11

[\[SWS\\_J1939Tp\\_00030\]](#) [\[SWS\\_J1939Tp\\_00180\]](#) [\[SWS\\_J1939Tp\\_00206\]](#)

### B.2 Traceable Item History of this Document According to AUTOSAR Release R24-11

#### B.2.1 Added Specification Items in R24-11

none

#### B.2.2 Changed Specification Items in R24-11

[\[SWS\\_J1939Tp\\_00041\]](#) [\[SWS\\_J1939Tp\\_00116\]](#)

#### B.2.3 Deleted Specification Items in R24-11

none