| | |
|---|---|
| **Document Title** | Specification of a Transport Layer for SAE J1939 |
| **Document Owner** | AUTOSAR |
| **Document Responsibility** | AUTOSAR |
| **Document Identification No** | 425 |

| | |
|---|---|
| **Document Status** | published |
| **Part of AUTOSAR Standard** | Classic Platform |
| **Part of Standard Release** | R24-11 |

**Document Change History**

| **Date** | **Release** | **Changed by** | **Description** |
|---|---|---|---|
| 2024-11-27 | R24-11 | AUTOSAR Release Management | • Changed lower layer to LSduR |
| 2023-11-23 | R23-11 | AUTOSAR Release Management | • Changed document name to include “CP”<br>• Added references to `ComMChannel`s to configuration<br>• Added information about automatic handle IDs to configuration |
| 2022-11-24 | R22-11 | AUTOSAR Release Management | • Fixed not applicable SWS items |
| 2021-11-25 | R21-11 | AUTOSAR Release Management | • No content changes |
| 2020-11-30 | R20-11 | AUTOSAR Release Management | • Improved structure of error sections<br>• Replaced error descriptions with generated tables |
| 2019-11-28 | R19-11 | AUTOSAR Release Management | • No content changes<br>• Changed Document Status from Final to published |

▽

△

| | | | |
|---|---|---|---|
| 2018-10-31 | 4.4.0 | AUTOSAR Release Management | • Changed header file structure<br>• Fixed reference to protocol type<br>• Harmonized J1939TP_E_UNINIT<br>• Clarifications of timeouts |
| 2017-12-08 | 4.3.1 | AUTOSAR Release Management | • Clarified reception with identical channels<br>• Introduction of runtime errors |
| 2016-11-30 | 4.3.0 | AUTOSAR Release Management | • Improved handling of meta data<br>• Reliable TxConfirmation replaces timeout<br>• Clarification on concurrent connections<br>• Support of link time configuration variant |
| 2015-07-31 | 4.2.2 | AUTOSAR Release Management | • Fixed retry behavior<br>• Clarified effect of MetaData on SA/DA/Protocol<br>• Introduction of further error classes |
| 2014-10-31 | 4.2.1 | AUTOSAR Release Management | • Removed obsolete configuration elements<br>• Harmonized with SWS BSW General |
| 2014-03-31 | 4.1.3 | AUTOSAR Release Management | • Allow for parallel reception via TP (BAM/CMDT) and directly<br>• Clarified ambiguities regarding CTS parameters |
| 2013-10-31 | 4.1.2 | AUTOSAR Release Management | • NotifResultType replaced by StdReturnType<br>• Additional development errors for protocol problems<br>• Connection parameters moved to channel configuration<br>• Editorial changes<br>• Removed chapter(s) on change documentation |

▽

△

| | | | |
|---|---|---|---|
| 2013-03-15 | 4.1.1 | AUTOSAR Administration | • Support for dynamic CAN IDs via MetaData<br>• Support for variable block size and retry<br>• TP API improvements and fixes<br>• Adapted to new BSW General |
| 2011-12-22 | 4.0.3 | AUTOSAR Administration | • API changes: `CancelTransmit`, `CancelReceive`, and `ChangeParameter` were changed to synchronous behavior |
| 2011-04-15 | 4.0.2 | AUTOSAR Administration | • Fixed service API IDs<br>• Removed duplicate requirement J1939TP0099<br>• Clarified requirements [SWS_J1939Tp_00125] and [SWS_J1939Tp_00189] |
| 2009-12-18 | 4.0.1 | AUTOSAR Administration | • Initial release |

## Disclaimer

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

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

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

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

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

# Contents

# 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 [1, SAE J1939]. The terms J1939Tp and 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. [2, 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 [2, 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 J1939 Transport Layer module, the reader of this specification needs to read also the original [2, 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:

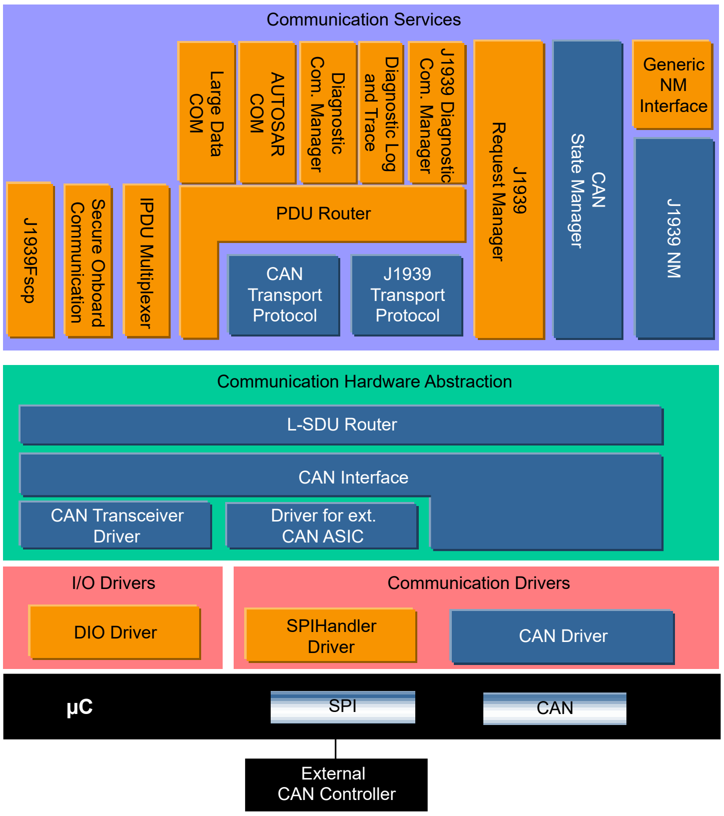

**Figure 1.1: AUTOSAR Communication Stack**

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 MetaData.

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 [3, AUTOSAR Glossary].

| Glossary Term | Explanation |
|---|---|
| Address Claiming | Address Claiming forms the network management of SAE J1939 defined in the standard document [4, SAE J1939-81]. Address claiming assigns a temporary 8-bit identifier to each ECU connected to one J1939 network. Within this network, the 8-bit identifier is unique. The 8-bit identifier is used as source and target address of Parameter Groups (messages) transferred via the J1939 network. The address claiming procedure is based on the exchange of AddressClaimed messages (PGN = 0x0EE00). |
| Parameter | A parameter is a signal of the SAE J1939 application layer. Parameters are uniquely identified by the SPN. |
| Parameter Group | A Parameter Group is a message of the SAE J1939 application layer. Each Parameter Group contains several Parameters (signals), and is uniquely identified by the PGN. |
| SAE J1939 Diagnostics | The SAE J1939 diagnostic layer is defined in the standard document [5, SAE J1939-73]. The J1939 diagnostics is functionally similar to the UDS diagnostics, and has recently been extended to support OBD for emission relevant values. |
| SAE J1939 Transport Protocol | The SAE J1939 transport protocol is used for the segmented transmission of messages with more than 8 bytes of data. The transport protocol is defined in the network layer standard document ([2, SAE J1939-21]). |

| Acronym / Abbreviation | Description |
|---|---|
| BAM | Broadcast Announce Message, broadcast variant of SAE J1939 transport protocol |
| BSW | Basic Software (module) |
| CanIf | CAN Interface |
| CanTp | CAN Transport Layer |
| CMDT | Connection Mode Data Transfer, a.k.a. RTS/CTS, peer-to-peer variant of SAE J1939 transport protocol |
| DA | Destination Address, part of the 29 bit identifier of SAE J1939 messages |
| DET | Default Error Tracer, supports development and runtime error reporting |
| DP | Data Page, the most significant bit (MSB) of the 18 bit PGN |
| EcuM | ECU State Manager |
| EDP | Extended Data Page, the second bit (after MSB) of the 18 bit PGN |
| FrTp | FlexRay Transport Layer |
| J1939Tp | SAE J1939 Transport Layer, implementing the SAE J1939 Transport Protocol |
| LSduR | L-SDU Router |
| MetaData | Meta data transferred alongside a PDU, consisting of a set of meta data items |
| MetaDataItem | A single item of meta data of defined type and size |
| PDU1 | J1939 PDU Type 1, this kind of PG can be sent to a specific destination address |
| PDU2 | J1939 PDU Type 2, this kind of PG can only be sent as broadcast |
| PDUF | PDU Format, the middle byte of the 18 bit PGN which identifies the PG and determines the layout (PDU1/PDU2) of the PGN |
| PduR | PDU Router |
| PDUS | PDU Specific, the lower byte of the 18 bit PGN which further identifies PDU2 PG which do cannot have a destination address |
| PG | Parameter Group, SAE J1939 term for a specific message layout, corresponds to an N-SDU of J1939Tp |

▽

△

| Acronym / Abbreviation | Description |
|---|---|
| PGN | Parameter Group Number, unique identifier (18 bits: EDP, DP, PDUF, PDUS) of an SAE J1939 Parameter Group that is contained in the payload of many J1939 protocol messages and in the 29bit CAN identifier of SAE J1939 messages. |
| RTE | AUTOSAR Runtime Environment |
| SA | Source Address, part of the 29 bit identifier of SAE J1939 messages |
| SchM | Basic Software Schedule Manager, part of the RTE |
| SNA | Signal Not Available, all bits set to 1 in SAE J1939 PGs/Parameters |
| SPN | Suspect Parameter Number, unique identifier of an SAE J1939 Parameter |
| TP.CM | Connection Management message (PGN = 0x0EC00) used by SAE J1939 transport protocol, corresponds to an N-PDU of J1939Tp |
| TP.CM_BAM | Broadcast Announce Message, variant of TP.CM that initiates a BAM transmission |
| TP.CM_CTS | Connection Mode Clear To Send, variant of TP.CM that is used for handshake during CMDT transmission |
| TP.CM_EndOfMsgAck | End Of Message Acknowledge, variant of TP.CM that acknowledges correct reception of a CMDT transmission |
| TP.CM_RTS | Connection Mode Request To Send, variant of TP.CM that initiates a CMDT transmission |
| TP.Conn_Abort | Connection Abort, variant of TP.CM that terminates a CMDT transmission |
| TP.DT | Data Transfer message (PGN = 0x0EB00) used by SAE J1939 transport protocol, corresponds to an N-PDU of J1939Tp |

# 3 Related Documentation

## 3.1 Input Documents & Related Standards and Norms

[1] SAE J1939 – Serial Control and Communications Heavy Duty Vehicle Network

[2] SAE J1939-21 Data Link Layer

[3] Glossary
AUTOSAR_FO_TR_Glossary

[4] SAE J1939-81 Network Management

[5] SAE J1939-73 Application Layer – Diagnostics

[6] General Specification of Basic Software Modules
AUTOSAR_CP_SWS_BSWGeneral

[7] Layered Software Architecture
AUTOSAR_CP_EXP_LayeredSoftwareArchitecture

[8] Specification of Default Error Tracer
AUTOSAR_CP_SWS_DefaultErrorTracer

[9] Specification of ECU State Manager
AUTOSAR_CP_SWS_ECUStateManager

[10] Specification of RTE Software
AUTOSAR_CP_SWS_RTE

[11] Specification of PDU Router
AUTOSAR_CP_SWS_PDURouter

[12] Specification of Linklayer Sdu Routing Module
AUTOSAR_CP_SWS_LSduRouter

[13] Specification of CAN Interface
AUTOSAR_CP_SWS_CANInterface

[14] Specification of CAN Transport Layer
AUTOSAR_CP_SWS_CANTransportLayer

[15] Requirements on BSW Modules for SAE J1939
AUTOSAR_CP_RS_SAEJ1939

[16] General Requirements on Basic Software Modules
AUTOSAR_CP_RS_BSWGeneral

[17] System Template
AUTOSAR_CP_TPS_SystemTemplate

[18] Specification of ECU Configuration
AUTOSAR_CP_TPS_ECUConfiguration

## 3.2 Related Specification

AUTOSAR provides a General Specification on Basic Software modules [6, SWS BSW General], which is also valid for SAE J1939 Transport Layer.

Thus, the specification [6, 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 J1939 Transport Layer supports the implementation of ECUs that are designed to operate in a J1939 network.

# 5 Dependencies to Other Modules

The [7, EXP Layered Software Architecture] shows an overview of the neighboring modules of the J1939 Transport Layer.

Besides the standard modules Default Error Tracer (DET, [8, SWS Default Error Tracer]), ECU State Manager (EcuM, [9, SWS ECU State Manager]), and Basic Software Scheduler (SchM, [10, SWS RTE]), which have interfaces to all BSW modules, J1939Tp only interacts with the PDU Router (PduR, [11, SWS PDU Router]) and the L-SDU Router (LSduR, [12, SWS L-SDU Router]), and indirectly with the CAN Interface (CanIf, [13, SWS CAN Interface]). The interfaces of J1939Tp are similar to the interfaces of CAN Transport Layer (CanTp, [14, SWS CAN Transport Layer]).

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.

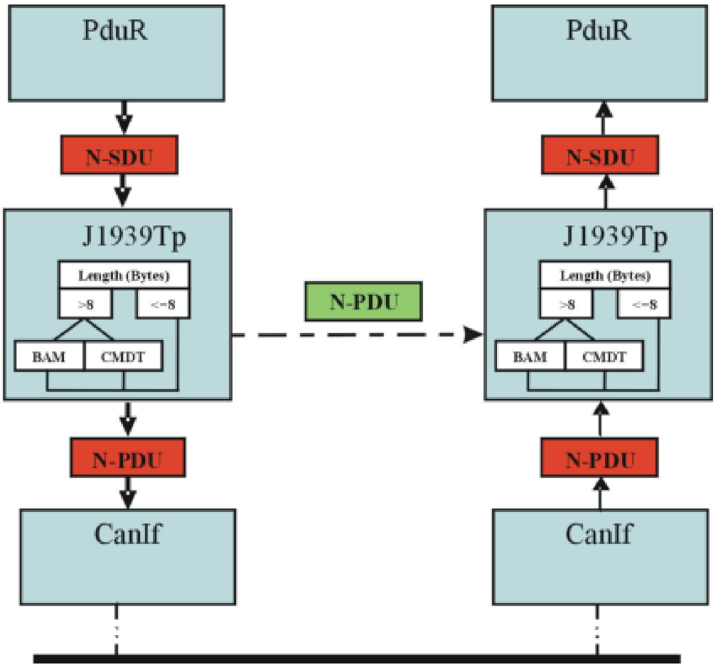

**Figure 5.1: J1939 Transport Layer interactions**

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 the subsection 5.1.6 “Code file structure” of the [6, SWS BSW General].

**[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 the subsection 5.1.7 “Header file structure” of the [6, SWS BSW General].

# 6 Requirements Tracing

The following tables reference the requirements specified in [15, SRS SAE J1939] (Requirements on BSW Modules for SAE J1939) and [16, 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_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] |
| **[SRS_BSW_00407]** | Each BSW module shall provide a function to read out the version information of a dedicated module implementation | [SWS_J1939Tp_00089] |

▽

△

| Requirement | Description | Satisfied by |
|---|---|---|
| **[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_J1939_00001]** | The J1939 Transport Layer module shall be configurable to support only transport protocol variant BAM | [SWS_J1939Tp_00125] |
| **[SRS_J1939_00010]** | The J1939 Transport Layer module shall implement an interface for initialization | [SWS_J1939Tp_00024] |
| **[SRS_J1939_00011]** | The J1939 Transport Layer services shall not be operational before initializing the module | [SWS_J1939Tp_00023] |
| **[SRS_J1939_00019]** | The J1939 Transport Layer module shall support the transport protocol variant BAM | [SWS_J1939Tp_00155] |
| **[SRS_J1939_00020]** | The AUTOSAR J1939 Transport Layer module shall support the transport protocol variant CMDT | [SWS_J1939Tp_00155] |
| **[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] |

**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 [2, 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 [2, 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]** ⌈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 J1939 Transport Layer shall implement the following two J1939 transport protocol variants defined in [2, SAE J1939-21]:

- BAM for broadcast transmission
- CMDT for point-to-point transmission

⌋

**[SWS_J1939Tp_00125]**

*Upstream requirements:* SRS_BSW_00171, SRS_J1939_00001

⌈The 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]** ⌈The 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_00231]** ⌈The 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 Section 7.5.7.

**[SWS_J1939Tp_00233]** ⌈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:

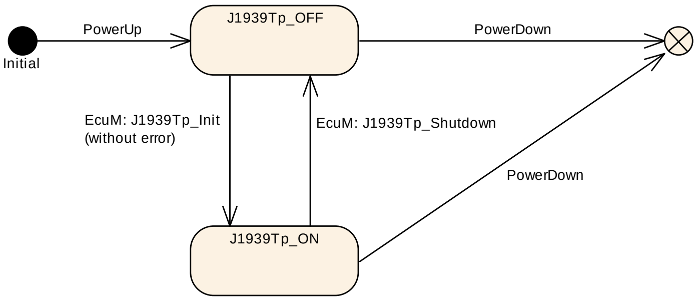

**Figure 7.1: J1939 Transport Layer life cycle**

**[SWS_J1939Tp_00019]**

*Upstream requirements:* SRS_BSW_00335, SRS_BSW_00441

⌈The J1939Tp module shall have two global states, J1939TP_ON and J1939TP_OFF.⌋

**[SWS_J1939Tp_00020]** ⌈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]** ⌈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_J1939NmTransmit.⌋

### 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 [2, 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 [2, 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,
- PduInfoPtr 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 MetaData 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 (NSduId) 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 MetaData 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 MetaData, these parameters are optional. If SA or DA is not configured, the value provided via the MetaData 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 NSduId and the SduLength provided in the PduInfoType structure. It shall not use the payload of N-SDUs, only the contained MetaData.⌋

**[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 Section 7.5.4.2 for a description how the 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.

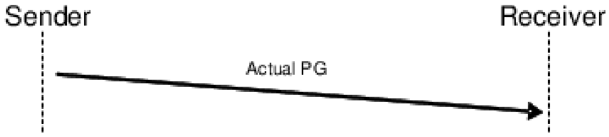

**Figure 7.2: Example of direct data flow for PGs of variable length $<=$ 8 Bytes**

### 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.

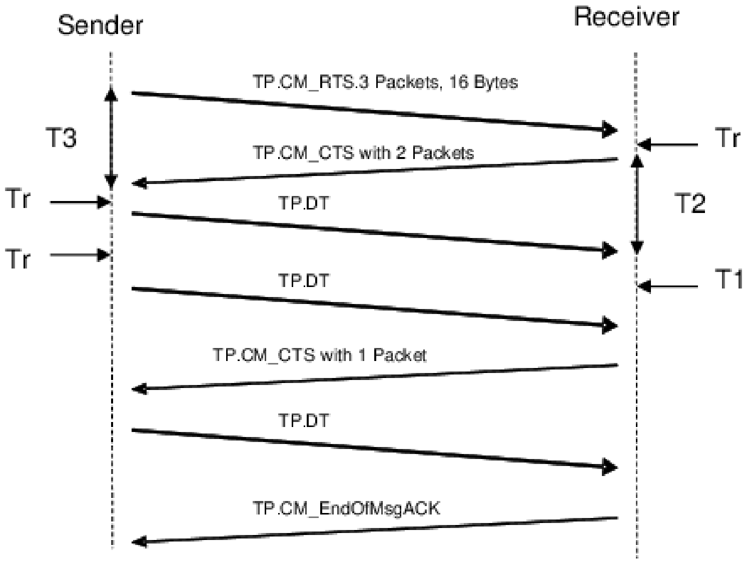

**Figure 7.3: Example of data flow without error using CMDT as transport protocol variant**

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 J1939Tp_Transmit() 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 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 [2, SAE J1939-21]) according to the timing requirements defined in [2, 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_CONFPENDING for the following calls.

⌋

**[SWS_J1939Tp_00195]** ⌈If a TP.CM_CTS wait frame (number of packets set to 0, see also [2, 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_DATARETRY 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 [2, SAE J1939-21]. They supervise the CMDT data flow.⌋

**[SWS_J1939Tp_00100]** ⌈If a timeout occurs during CMDT transmission (see [2, 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 [2, 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, Tr, T1) as described in [2, 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 [2, 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.

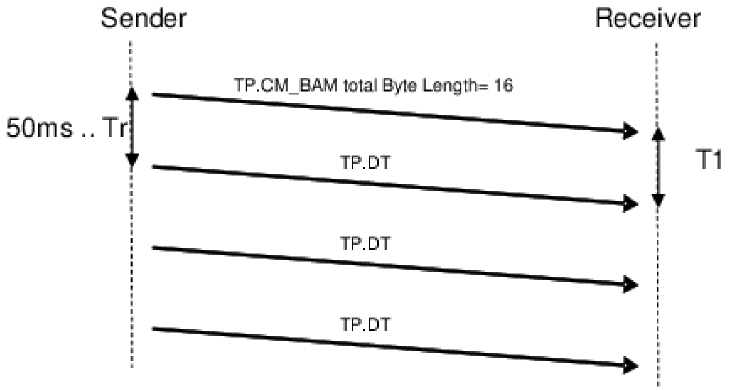

**Figure 7.4: Example of data flow using BAM as transport protocol variant**

### 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():

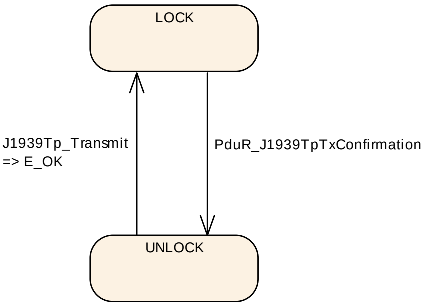

**Figure 7.5: Tx Buffer locking**

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

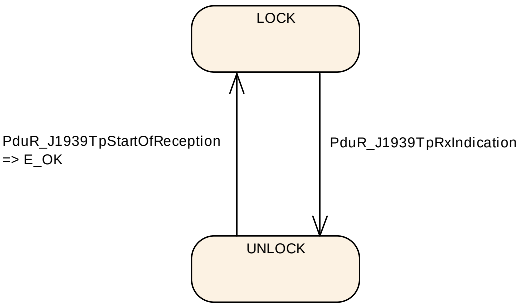

**Figure 7.6: Rx Buffer locking**

### 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.

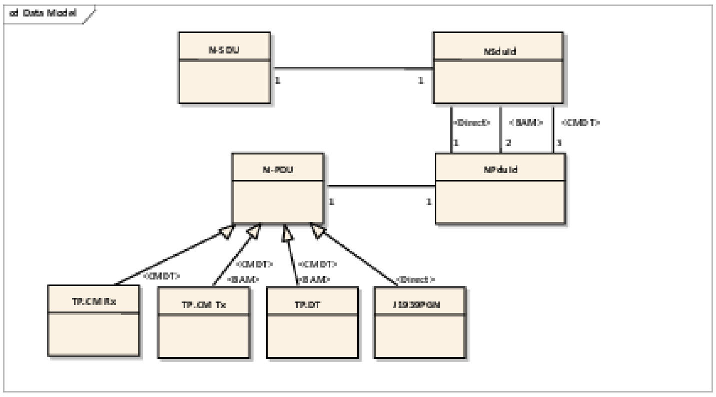

**Figure 7.7: Relation between N-SDU and N-PDU**

The N-PDUs as well as the N-SDU may use MetaData. 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 MetaData, from the addressing information implicitly encoded in the NPduIds or explicitly provided via the MetaData 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 MetaData, or by the SA/DA explicitly provided in the MetaData 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:

- sent from a given SA - in the BAM case
- sent from a given SA to a given DA - in the CMDT case

This reduces 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 [2, 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.

### 7.5.8 N-PDU Padding

**[SWS_J1939Tp_00200]** ⌈The J1939 Transport Layer module shall send TP.DT frames always with 8 bytes according to [2, SAE J1939-21].⌋

**[SWS_J1939Tp_00068]**

*Upstream requirements:* SRS_J1939_00024

⌈According to [2, 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

The section 7.2 “Error Handling” of the [6, SWS BSW General] 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] Definiton 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] Definiton 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 |

▽

△

| *Type of error* | *Related error code* | *Error value* |
|---|---|---|
| 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 [6, SWS BSW General] by [SWS_BSW_00212].

**[SWS_J1939Tp_00188]**

*Upstream requirements:* SRS_BSW_00323

⌈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 | TPParameterType |
| | 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 | |
| | ***Type*** | – |
| | ***Comment*** | The content of the initialization data structure is implementation specific. |
| ***Description*** | Data structure containing post-build configuration data of J1939-TP. | |
| ***Available via*** | 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

⌈

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_Init | |
| ***Syntax*** | `void J1939Tp_Init (`<br>`  const J1939Tp_ConfigType* ConfigPtr`<br>`)` | |
| ***Service ID [hex]*** | 0x01 | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Non Reentrant | |
| ***Parameters (in)*** | ConfigPtr | Pointer to configuration data structure. |
| ***Parameters (inout)*** | None | |
| ***Parameters (out)*** | None | |
| ***Return value*** | None | |
| ***Description*** | This function initializes the J1939Tp module. | |
| ***Available via*** | 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]** ⌈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 [6, 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

⌈

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

⌈

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_GetVersionInfo | |
| ***Syntax*** | void J1939Tp_GetVersionInfo (<br>  Std_VersionInfoType* VersionInfo<br>) | |
| ***Service ID [hex]*** | 0x03 | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Reentrant | |
| ***Parameters (in)*** | None | |
| ***Parameters (inout)*** | None | |
| ***Parameters (out)*** | VersionInfo | Pointer to the location where the version information of J1939Tp shall be stored. |
| ***Return value*** | None | |
| ***Description*** | Returns the version information of J1939Tp. | |
| ***Available via*** | J1939Tp.h | |

⌋

See subsection 8.3.4 “Get Version Information” of [6, SWS BSW General] for details. The module ID of the SAE J1939 Transport Layer is also defined in [6, SWS BSW General].

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

⌈

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_Transmit | |
| ***Syntax*** | `Std_ReturnType J1939Tp_Transmit (`<br>`  PduIdType TxPduId,`<br>`  const PduInfoType* PduInfoPtr`<br>`)` | |
| ***Service ID [hex]*** | 0x49 | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Reentrant for different PduIds. Non reentrant for the same PduId. | |
| ***Parameters (in)*** | TxPduId | Identifier of the PDU to be transmitted |
| | PduInfoPtr | Length of and pointer to the PDU data and pointer to MetaData. |
| ***Parameters (inout)*** | None | |
| ***Parameters (out)*** | None | |
| ***Return value*** | Std_ReturnType | E_OK: Transmit request has been accepted.<br>E_NOT_OK: Transmit request has not been accepted. |
| ***Description*** | Requests transmission of a PDU. | |
| ***Available via*** | 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 J1939 Transport Layer session.⌋

**[SWS_J1939Tp_00030]** ⌈The function J1939Tp_Transmit() shall reject the transmit request and return the status value E_NOT_OK if the transmission needs a transport protocol and the channel is occupied.⌋

The term channel refers to a communication relation with identical SA and DA. See also Section 7.5.7 for further information on concurrent connections.

### 8.4.5 J1939Tp_CancelTransmit

#### [SWS_J1939Tp_00177] Definition of API function J1939Tp_CancelTransmit ⌈

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

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_CancelReceive | |
| ***Syntax*** | `Std_ReturnType J1939Tp_CancelReceive (`<br>`  PduIdType RxPduId`<br>`)` | |
| ***Service ID [hex]*** | 0x4c | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Non Reentrant | |
| ***Parameters (in)*** | RxPduId | Identification of the PDU to be cancelled. |
| ***Parameters (inout)*** | None | |

▽

△

| | | |
|---|---|---|
| ***Parameters (out)*** | None | |
| ***Return value*** | Std_ReturnType | E_OK: Cancellation was executed successfully by the destination module.<br>E_NOT_OK: Cancellation was rejected by the destination module. |
| ***Description*** | Requests cancellation of an ongoing reception of a PDU in a lower layer transport protocol module. | |
| ***Available via*** | 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.⌋

### 8.4.7 J1939Tp_ChangeParameter

**[SWS_J1939Tp_00180] Definition of API function J1939Tp_ChangeParameter** ⌈

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_ChangeParameter | |
| ***Syntax*** | `Std_ReturnType J1939Tp_ChangeParameter (`<br>`  PduIdType id,`<br>`  TPParameterType parameter,`<br>`  uint16 value`<br>`)` | |
| ***Service ID [hex]*** | 0x4b | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Non Reentrant | |
| ***Parameters (in)*** | id | Identification of the PDU which the parameter change shall affect. |
| | parameter | ID of the parameter that shall be changed. |
| | value | The new value of the parameter. |
| ***Parameters (inout)*** | None | |
| ***Parameters (out)*** | None | |
| ***Return value*** | Std_ReturnType | E_OK: The parameter was changed successfully.<br>E_NOT_OK: The parameter change was rejected. |
| ***Description*** | Request to change a specific transport protocol parameter (e.g. block size). | |
| ***Available via*** | J1939Tp.h | |

⌋

**[SWS_J1939Tp_00206]** ⌈J1939Tp_ChangeParameter() shall return E_NOT_OK if id is invalid, if parameter is not TP_BS, or if value is larger than 255.⌋

## 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

⌈

| | | |
|---|---|---|
| ***Service Name*** | J1939Tp_RxIndication | |
| ***Syntax*** | `void J1939Tp_RxIndication (`<br>`  PduIdType RxPduId,`<br>`  const PduInfoType* PduInfoPtr`<br>`)` | |
| ***Service ID [hex]*** | 0x42 | |
| ***Sync/Async*** | Synchronous | |
| ***Reentrancy*** | Reentrant for different PduIds. Non reentrant for the same PduId. | |
| ***Parameters (in)*** | 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. |
| ***Parameters (inout)*** | None | |
| ***Parameters (out)*** | None | |
| ***Return value*** | None | |
| ***Description*** | Indication of a received PDU from a lower layer communication interface module. | |
| ***Available via*** | 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

⌈

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

⌈

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

⌋

**[SWS_J1939Tp_00106]** ⌈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. |
| 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. |

▽

△

| *API Function* | *Header File* | *Description* |
|---|---|---|
| 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_DATARETRY. 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 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 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.

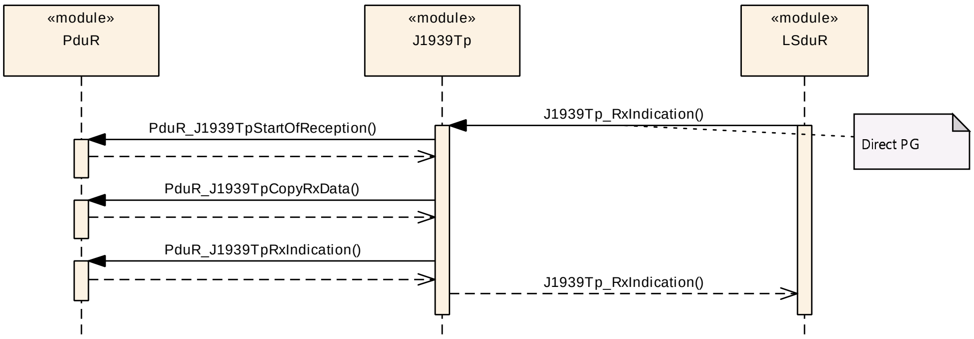

**Figure 9.1: Reception of Direct PG**

## 9.2 Reception via BAM

The following diagram shows the interaction of the 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.

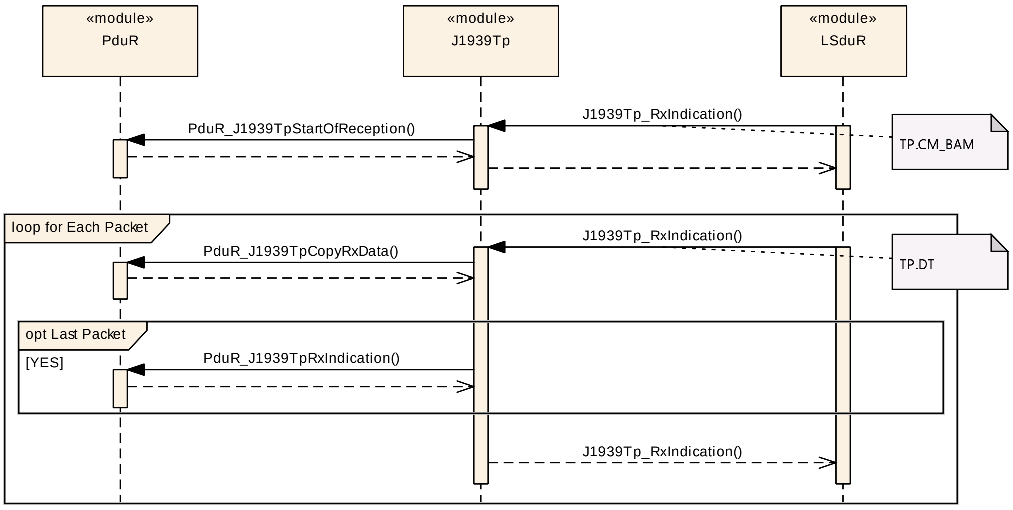

**Figure 9.2: Reception via BAM**

## 9.3 Reception via CMDT

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

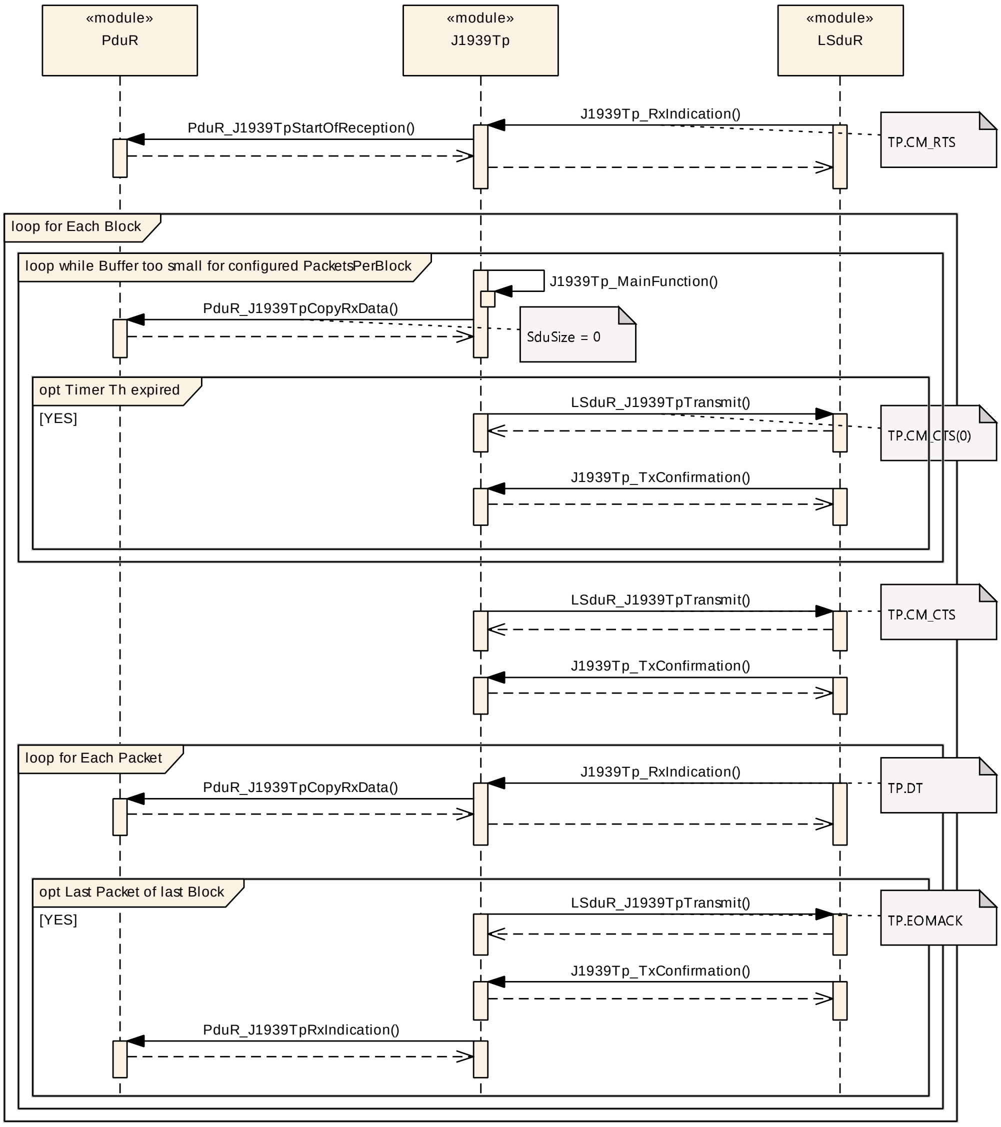

**Figure 9.3: Reception via CMDT**

## 9.4 Transmission of Direct PG

The following diagram shows the interaction of the 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.

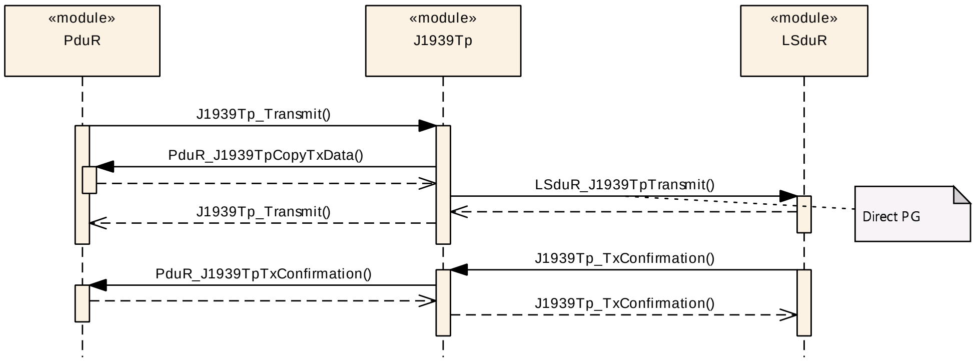

**Figure 9.4: Transmission of Direct PG**

## 9.5 Transmission via BAM

The following diagram shows the interaction of the 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.

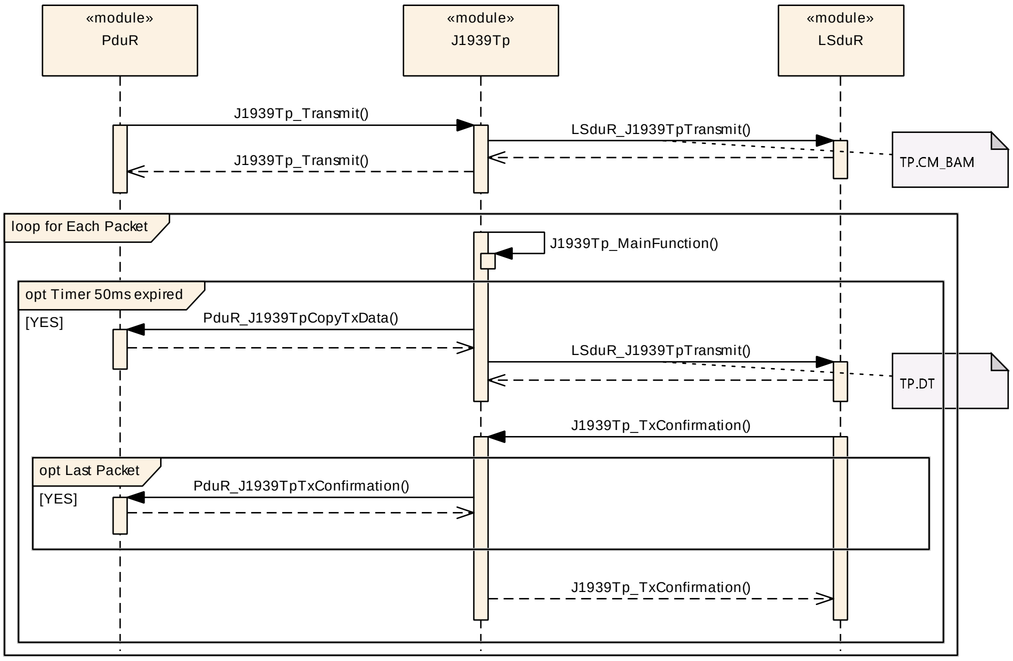

**Figure 9.5: Transmission via BAM**

## 9.6 Transmission via CMDT

The following diagram shows the interaction of the 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.

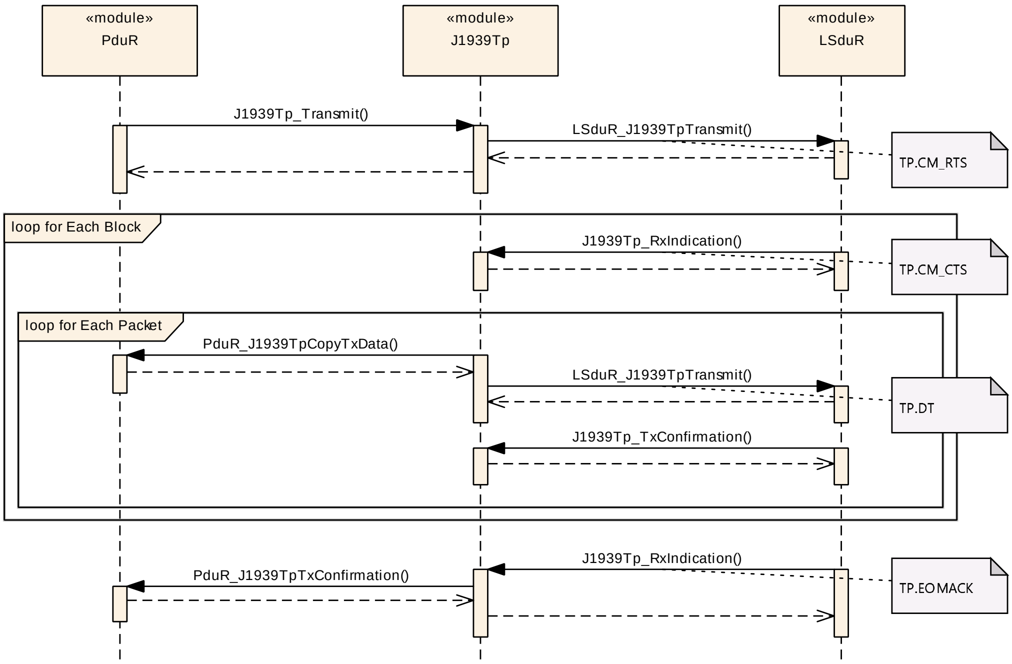

**Figure 9.6: Transmission via CMDT**

## 9.7 Handling of Retry during CMDT Transmission

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

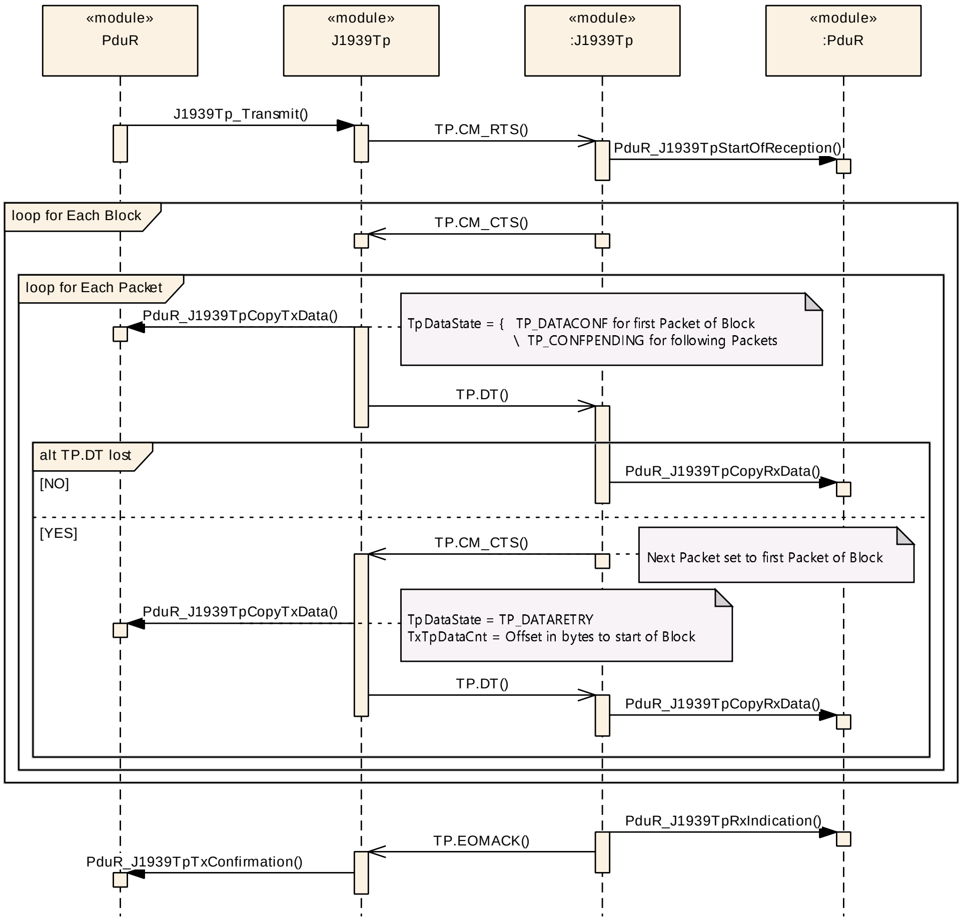

**Figure 9.7: Retry Handling**

# 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 the section 10.1 “Introduction to configuration specification” in [6, SWS BSW General]. For details about published information of the J1939 Transport Layer module, refer to the section 10.3 “Published Information” in [6, SWS BSW General].

In Section 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 [17, TPS System Template], which also contains the rules for these derivations.

The following pictures show an overview of the configuration parameters available for J1939Tp:

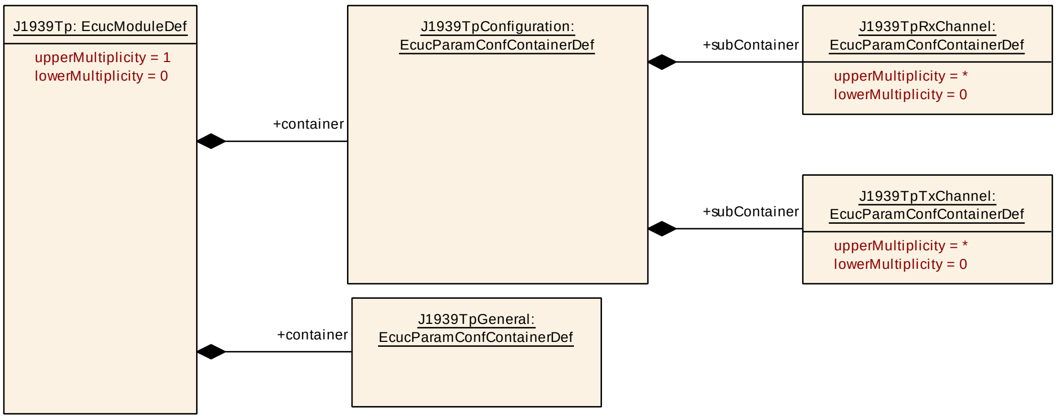

**Figure 10.1: Module Configuration**

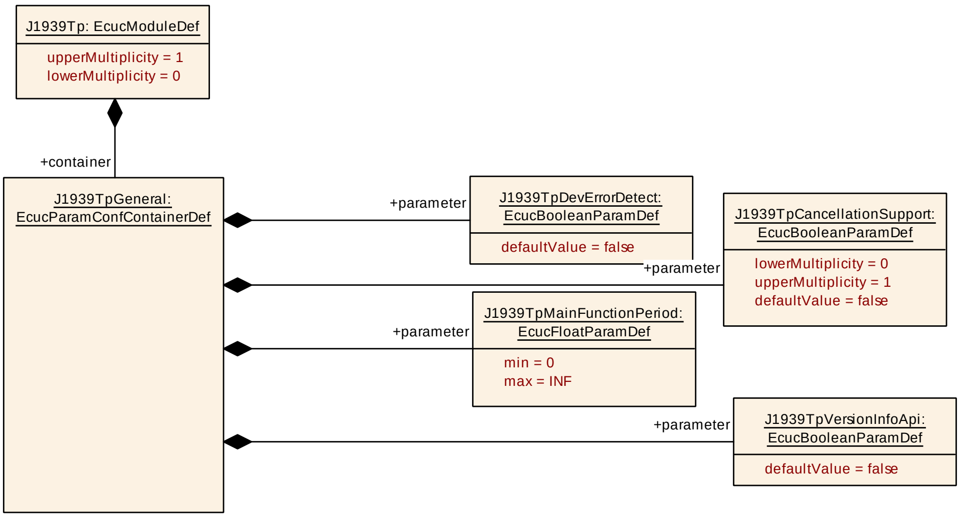

**Figure 10.2: General Parameters**

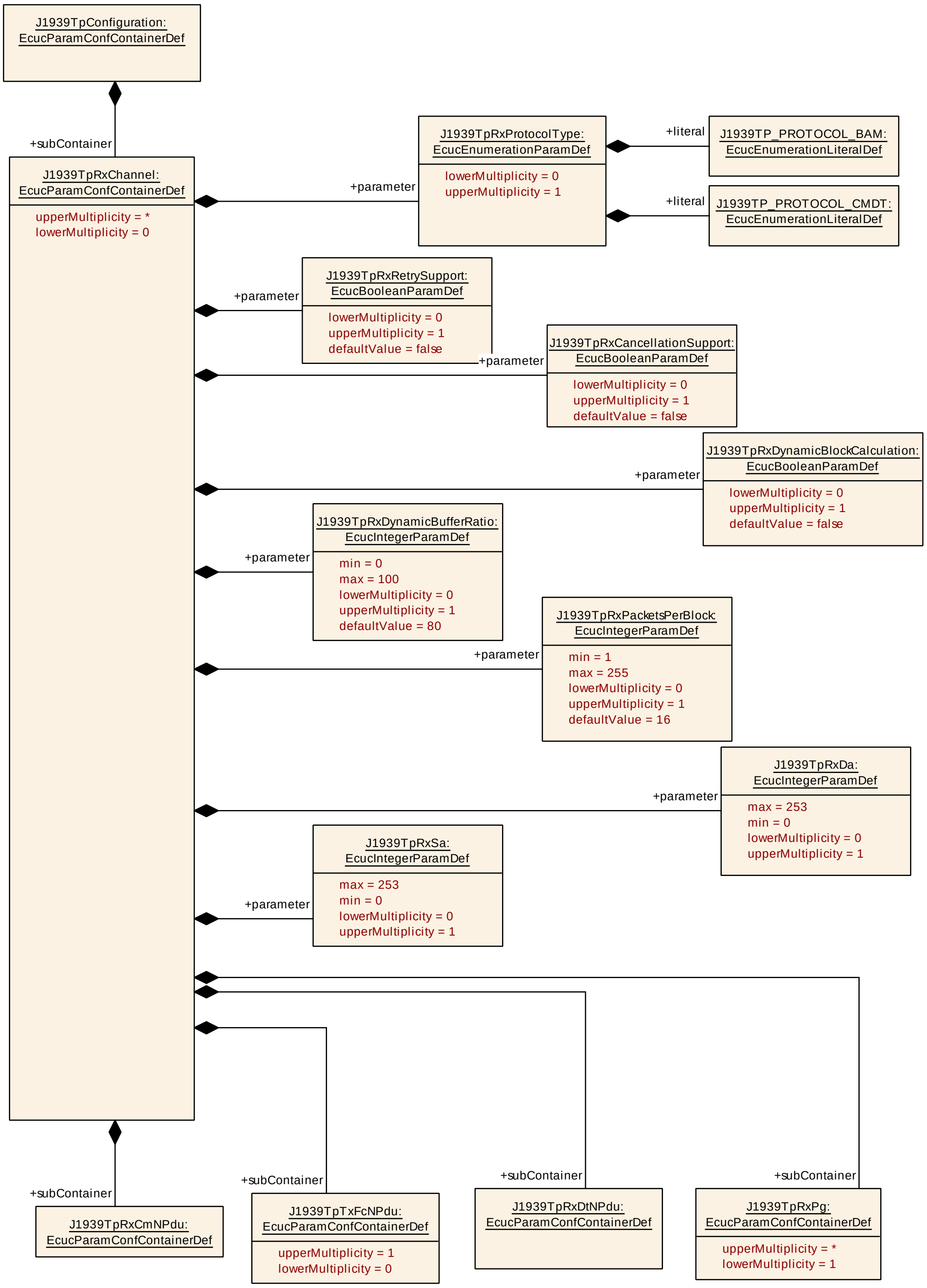

**Figure 10.3: Configuration of Rx Channel - Part 1**

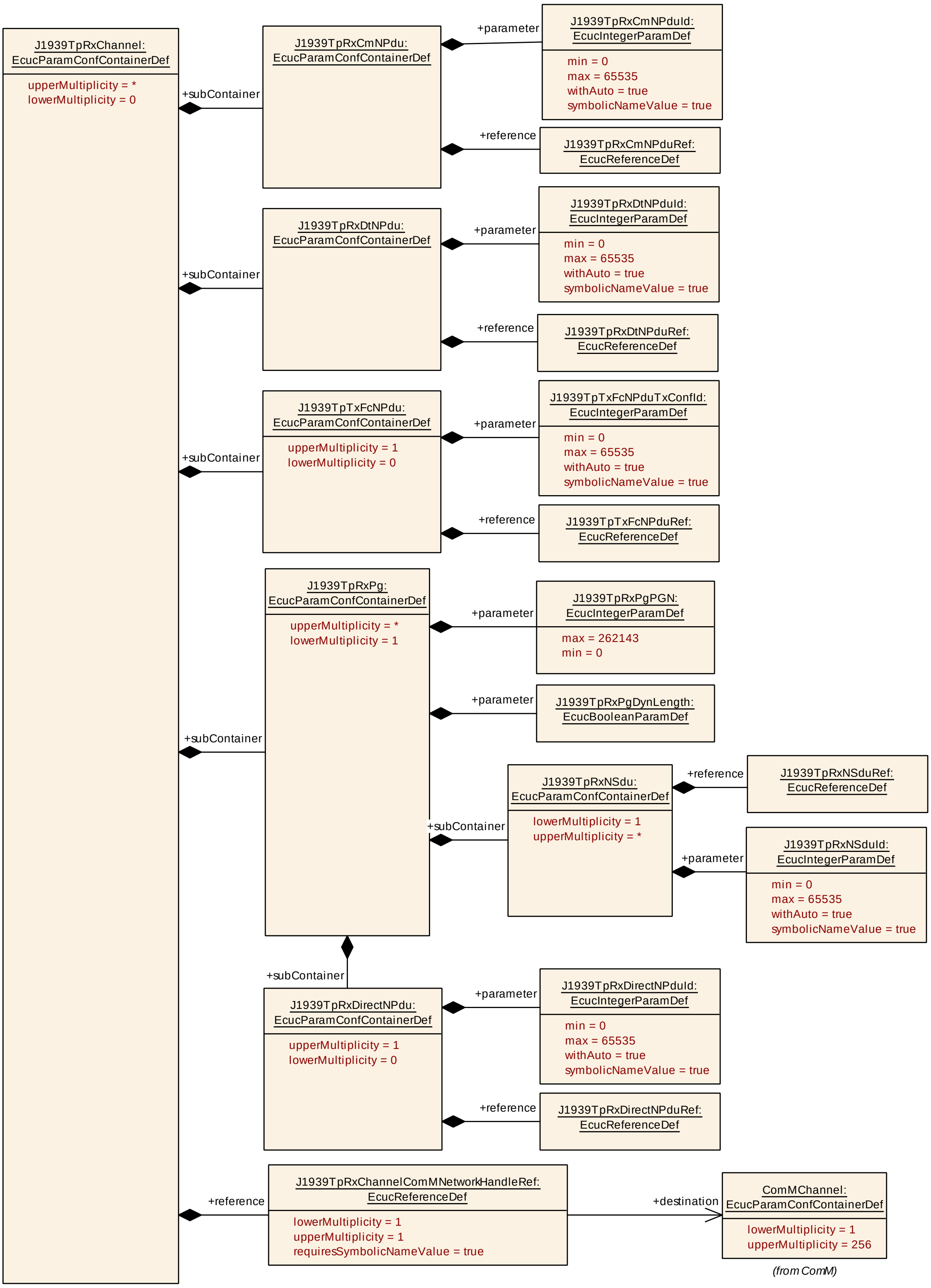
J1939TpRxChannel: EcucParamConfContainerDef
upperMultiplicity = *
lowerMultiplicity = 0
+subContainer
J1939TpRxCmNPdu: EcucParamConfContainerDef
+parameter
J1939TpRxCmNPduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpRxCmNPduRef: EcucReferenceDef
+subContainer
J1939TpRxDtNPdu: EcucParamConfContainerDef
+parameter
J1939TpRxDtNPduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpRxDtNPduRef: EcucReferenceDef
+subContainer
J1939TpTxFcNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
+parameter
J1939TpTxFcNPduTxConfId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpTxFcNPduRef: EcucReferenceDef
+subContainer
J1939TpRxPg: EcucParamConfContainerDef
upperMultiplicity = *
lowerMultiplicity = 1
+parameter
J1939TpRxPgPGN: EcucIntegerParamDef
max = 262143
min = 0
+parameter
J1939TpRxPgDynLength: EcucBooleanParamDef
+subContainer
J1939TpRxNSdu: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = *
+reference
J1939TpRxNSduRef: EcucReferenceDef
+parameter
J1939TpRxNSduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+subContainer
J1939TpRxDirectNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
+parameter
J1939TpRxDirectNPduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpRxDirectNPduRef: EcucReferenceDef
+reference
J1939TpRxChannelComMNetworkHandleRef: EcucReferenceDef
lowerMultiplicity = 1
upperMultiplicity = 1
requiresSymbolicNameValue = true
+destination
ComMChannel: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = 256
(from ComM)

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

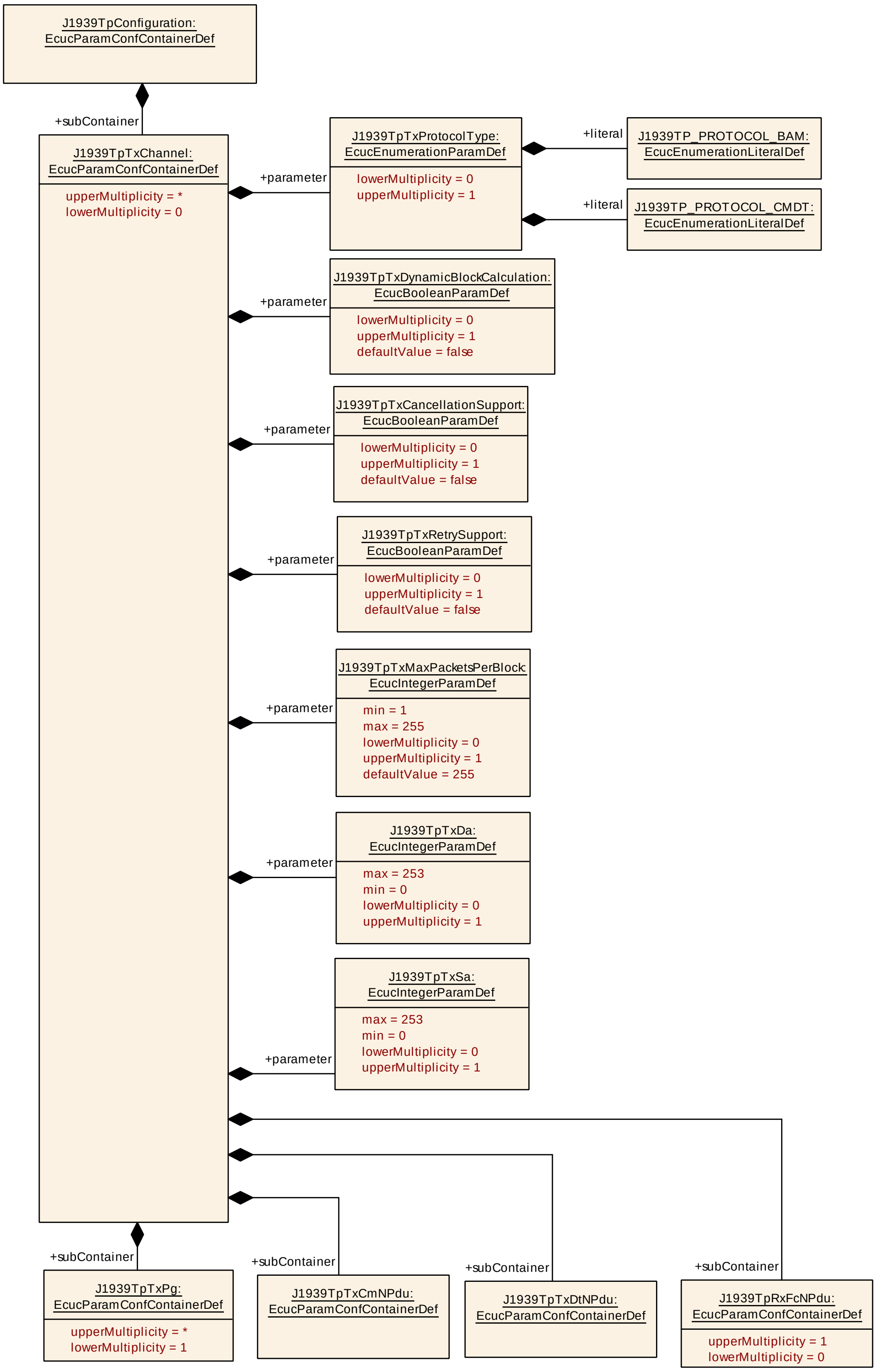

**Figure 10.5: Configuration of Tx Channel - Part 1**

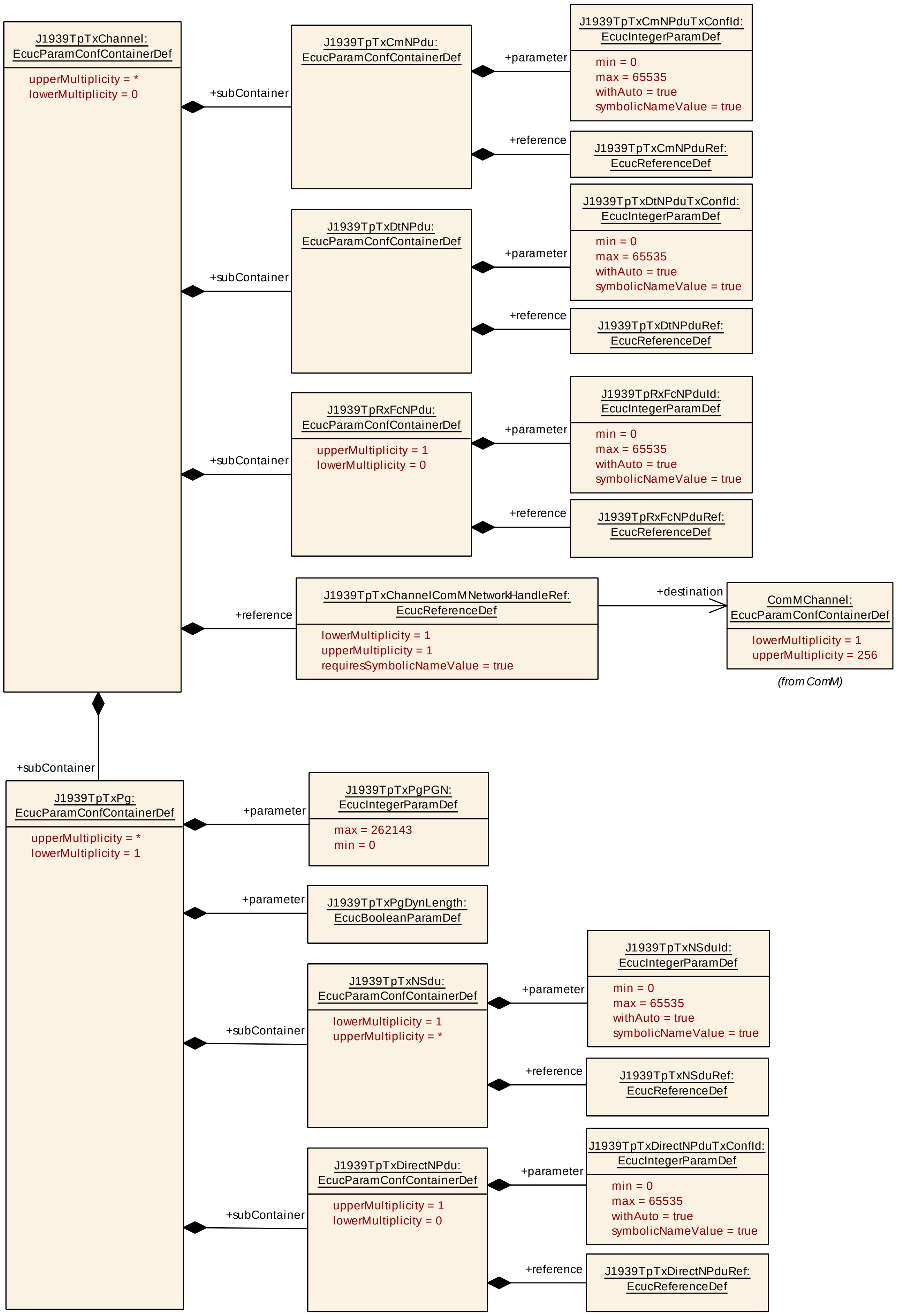
J1939TpTxChannel: EcucParamConfContainerDef
upperMultiplicity = *
lowerMultiplicity = 0
+subContainer
J1939TpTxCmNPdu: EcucParamConfContainerDef
+parameter
J1939TpTxCmNPduTxConfId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpTxCmNPduRef: EcucReferenceDef
+subContainer
J1939TpTxDtNPdu: EcucParamConfContainerDef
+parameter
J1939TpTxDtNPduTxConfId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpTxDtNPduRef: EcucReferenceDef
+subContainer
J1939TpRxFcNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
+parameter
J1939TpRxFcNPduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpRxFcNPduRef: EcucReferenceDef
+reference
J1939TpTxChannelComMNetworkHandleRef: EcucReferenceDef
lowerMultiplicity = 1
upperMultiplicity = 1
requiresSymbolicNameValue = true
+destination
ComMChannel: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = 256
(from ComM)
+subContainer
J1939TpTxPg: EcucParamConfContainerDef
upperMultiplicity = *
lowerMultiplicity = 1
+parameter
J1939TpTxPgPGN: EcucIntegerParamDef
max = 262143
min = 0
+parameter
J1939TpTxPgDynLength: EcucBooleanParamDef
+subContainer
J1939TpTxNSdu: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = *
+parameter
J1939TpTxNSduId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpTxNSduRef: EcucReferenceDef
+subContainer
J1939TpTxDirectNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
+parameter
J1939TpTxDirectNPduTxConfId: EcucIntegerParamDef
min = 0
max = 65535
withAuto = true
symbolicNameValue = true
+reference
J1939TpTxDirectNPduRef: EcucReferenceDef

**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 [18, TPS ECU Configuration].

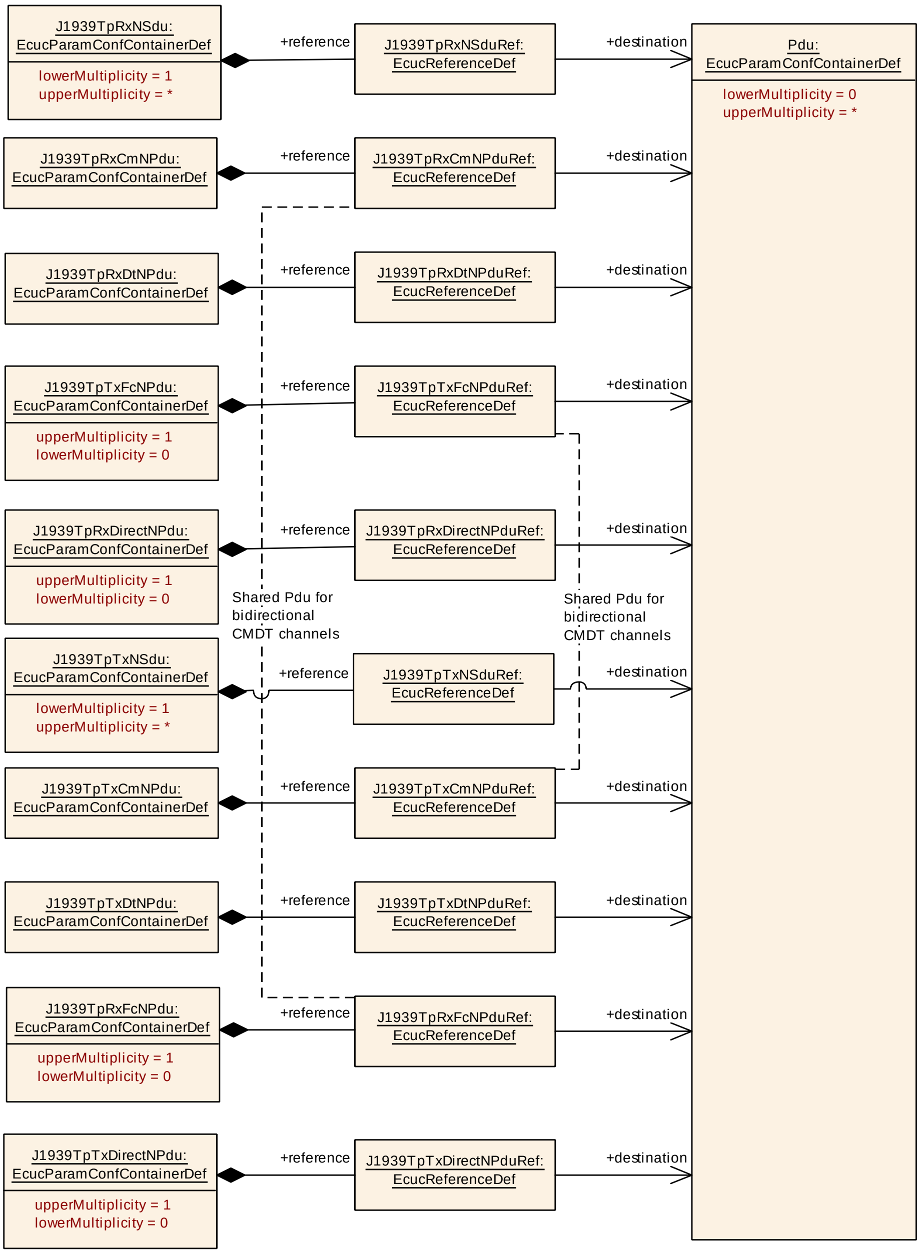
J1939TpRxNSdu: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = *
+reference
J1939TpRxNSduRef: EcucReferenceDef
+destination
Pdu: EcucParamConfContainerDef
lowerMultiplicity = 0
upperMultiplicity = *
J1939TpRxCmNPdu: EcucParamConfContainerDef
J1939TpRxCmNPduRef: EcucReferenceDef
J1939TpRxDtNPdu: EcucParamConfContainerDef
J1939TpRxDtNPduRef: EcucReferenceDef
J1939TpTxFcNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
J1939TpTxFcNPduRef: EcucReferenceDef
J1939TpRxDirectNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
J1939TpRxDirectNPduRef: EcucReferenceDef
Shared Pdu for bidirectional CMDT channels
Shared Pdu for bidirectional CMDT channels
J1939TpTxNSdu: EcucParamConfContainerDef
lowerMultiplicity = 1
upperMultiplicity = *
J1939TpTxNSduRef: EcucReferenceDef
J1939TpTxCmNPdu: EcucParamConfContainerDef
J1939TpTxCmNPduRef: EcucReferenceDef
J1939TpTxDtNPdu: EcucParamConfContainerDef
J1939TpTxDtNPduRef: EcucReferenceDef
J1939TpRxFcNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
J1939TpRxFcNPduRef: EcucReferenceDef
J1939TpTxDirectNPdu: EcucParamConfContainerDef
upperMultiplicity = 1
lowerMultiplicity = 0
J1939TpTxDirectNPduRef: EcucReferenceDef

**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** | **Scope / Dependency** |
| J1939TpConfiguration | 1 | This container contains the configuration parameters and sub containers of the J1939Tp module that define the communication paths. |
| J1939TpGeneral | 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** | J1939Tp |
| **Description** | This container describes the general configuration parameters of the J1939Tp module. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpCancellationSupport | 0..1 | [ECUC_J1939Tp_00174] |
| J1939TpDevErrorDetect | 1 | [ECUC_J1939Tp_00042] |
| J1939TpMainFunctionPeriod | 1 | [ECUC_J1939Tp_00044] |
| J1939TpVersionInfoApi | 1 | [ECUC_J1939Tp_00051] |

**No Included Containers**

⌋

**[ECUC_J1939Tp_00174] Definition of EcucBooleanParamDef J1939TpCancellationSupport ⌈**

| **Parameter Name** | J1939TpCancellationSupport |
|---|---|
| **Parent Container** | J1939TpGeneral |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00042] Definition of EcucBooleanParamDef J1939TpDevErrorDetect ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpDevErrorDetect | | |
| **Parent Container** | J1939TpGeneral | | |
| **Description** | Switches the development error detection and notification on or off.<br>• true: detection and notification is enabled.<br>• false: detection and notification is disabled. | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00044] Definition of EcucFloatParamDef J1939TpMainFunctionPeriod ⌈

| | | |
|---|---|---|
| **Parameter Name** | J1939TpMainFunctionPeriod | |
| **Parent Container** | J1939TpGeneral | |
| **Description** | Allow to configure the time for the MainFunction (in seconds). Please note: This configuration value shall be equal to the value in the ScheduleManager 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** | – | |
| **Scope / Dependency** | scope: ECU | | |

⌋

### [ECUC_J1939Tp_00051] Definition of EcucBooleanParamDef J1939TpVersionInfoApi ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpVersionInfoApi | | |
| **Parent Container** | J1939TpGeneral | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.3 J1939TpConfiguration

### [ECUC_J1939Tp_00052] Definition of EcucParamConfContainerDef J1939TpConfiguration ⌈

| | |
|---|---|
| **Container Name** | J1939TpConfiguration |
| **Parent Container** | J1939Tp |
| **Description** | This container contains the configuration parameters and sub containers of the J1939Tp module that define the communication paths. |
| **Configuration Parameters** | |

| **No Included Parameters** |
|---|

| Included Containers | | |
|---|---|---|
| **Container Name** | **Multiplicity** | **Scope / Dependency** |
| J1939TpRxChannel | 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. |
| J1939TpTxChannel | 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** | J1939TpConfiguration |
| **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. |
| **Configuration Parameters** | |

| Included Parameters | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxCancellationSupport | 0..1 | [ECUC_J1939Tp_00186] |
| J1939TpRxDa | 0..1 | [ECUC_J1939Tp_00178] |
| J1939TpRxDynamicBlockCalculation | 0..1 | [ECUC_J1939Tp_00187] |
| J1939TpRxDynamicBufferRatio | 0..1 | [ECUC_J1939Tp_00188] |
| J1939TpRxPacketsPerBlock | 0..1 | [ECUC_J1939Tp_00189] |
| J1939TpRxProtocolType | 0..1 | [ECUC_J1939Tp_00029] |
| J1939TpRxRetrySupport | 0..1 | [ECUC_J1939Tp_00185] |
| J1939TpRxSa | 0..1 | [ECUC_J1939Tp_00179] |
| J1939TpRxChannelComMNetworkHandleRef | 1 | [ECUC_J1939Tp_00194] |

| Included Containers | | |
|---|---|---|
| **Container Name** | **Multiplicity** | **Scope / Dependency** |
| J1939TpRxCmNPdu | 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. |
| J1939TpRxDtNPdu | 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. |
| J1939TpRxPg | 1..* | Parameter group received by the J1939 transport layer. |
| J1939TpTxFcNPdu | 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.<br>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** | J1939TpRxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00178] Definition of EcucIntegerParamDef J1939TpRxDa ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDa | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00187] Definition of EcucBooleanParamDef J1939TpRxDynamicBlockCalculation ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDynamicBlockCalculation | | |
| **Parent Container** | J1939TpRxChannel | | |
| **Description** | Enable dynamic calculation of "number of packets that can be sent" value in TP.CM_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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00188] Definition of EcucIntegerParamDef J1939TpRxDynamicBufferRatio ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDynamicBufferRatio | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local<br>dependency: Only applicable when "J1939TpRxRetrySupport" and "J1939TpRxDynamicBlockCalculation" are enabled | | |

⌋

### [ECUC_J1939Tp_00189] Definition of EcucIntegerParamDef J1939TpRxPacketsPerBlock ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxPacketsPerBlock | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00029] Definition of EcucEnumerationParamDef J1939TpRxProtocolType ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxProtocolType | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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).<br>This protocol uses two N-PDUs: The CmNPdu and the DtNPdu. | |
| | J1939TP_PROTOCOL_CMDT | J1939 transport protocol type CMDT (Connection Mode Data Transfer).<br>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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00185] Definition of EcucBooleanParamDef J1939TpRxRetrySupport ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxRetrySupport | | |
| **Parent Container** | J1939TpRxChannel | | |
| **Description** | Enable support for triggering repetition of failed transmission using TP.CM_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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00179] Definition of EcucIntegerParamDef J1939TpRxSa ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxSa | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00194] Definition of EcucReferenceDef J1939TpRxChannelComMNetworkHandleRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxChannelComMNetworkHandleRef | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.5 J1939TpRxCmNPdu

### [ECUC_J1939Tp_00128] Definition of EcucParamConfContainerDef J1939TpRxCmNPdu ⌈

| | | | |
|---|---|---|---|
| **Container Name** | J1939TpRxCmNPdu | | |
| **Parent Container** | J1939TpRxChannel | | |
| **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 consumes a meta data item of type CAN_ID_32. | | |
| **Configuration Parameters** | | | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxCmNPduId | 1 | [ECUC_J1939Tp_00129] |
| J1939TpRxCmNPduRef | 1 | [ECUC_J1939Tp_00158] |

**No Included Containers**

⌋

## [ECUC_J1939Tp_00129] Definition of EcucIntegerParamDef J1939TpRxCmNPduId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxCmNPduId | | |
| **Parent Container** | J1939TpRxCmNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

## [ECUC_J1939Tp_00158] Definition of EcucReferenceDef J1939TpRxCmNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxCmNPduRef | | |
| **Parent Container** | J1939TpRxCmNPdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### 10.1.6 J1939TpRxDtNPdu

**[ECUC_J1939Tp_00117] Definition of EcucParamConfContainerDef J1939TpRxDtNPdu ⌈**

| | |
|---|---|
| **Container Name** | J1939TpRxDtNPdu |
| **Parent Container** | J1939TpRxChannel |
| **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 consumes a meta data item of type CAN_ID_32. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxDtNPduId | 1 | [ECUC_J1939Tp_00133] |
| J1939TpRxDtNPduRef | 1 | [ECUC_J1939Tp_00134] |

| **No Included Containers** |
|---|

⌋

**[ECUC_J1939Tp_00133] Definition of EcucIntegerParamDef J1939TpRxDtNPduId ⌈**

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDtNPduId | | |
| **Parent Container** | J1939TpRxDtNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00134] Definition of EcucReferenceDef J1939TpRxDtNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDtNPduRef | | |
| **Parent Container** | J1939TpRxDtNPdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.7 J1939TpRxPg

### [ECUC_J1939Tp_00050] Definition of EcucParamConfContainerDef J1939TpRxPg ⌈

| | |
|---|---|
| **Container Name** | J1939TpRxPg |
| **Parent Container** | J1939TpRxChannel |
| **Description** | Parameter group received by the J1939 transport layer. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxPgDynLength | 1 | [ECUC_J1939Tp_00066] |
| J1939TpRxPgPGN | 1 | [ECUC_J1939Tp_00065] |

| **Included Containers** | | |
|---|---|---|
| **Container Name** | **Multiplicity** | **Scope / Dependency** |
| J1939TpRxDirectNPdu | 0..1 | This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less that 8 bytes. This N-PDU consumes a meta data item of type CAN_ID_32.<br>Please note: This sub container is only necessary when J1939TpRxPgDynLength is TRUE. |
| J1939TpRxNSdu | 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** | J1939TpRxPg | | |
| **Description** | This flag is set to TRUE when the N-SDU refers to a PGN with variable length.<br>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** | – | |
| **Scope / Dependency** | | | |

⌋

### [ECUC_J1939Tp_00065] Definition of EcucIntegerParamDef J1939TpRxPgPGN ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxPgPGN | | |
| **Parent Container** | J1939TpRxPg | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.8 J1939TpRxDirectNPdu

### [ECUC_J1939Tp_00130] Definition of EcucParamConfContainerDef J1939TpRxDirectNPdu ⌈

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

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxDirectNPduId | 1 | [ECUC_J1939Tp_00131] |
| J1939TpRxDirectNPduRef | 1 | [ECUC_J1939Tp_00132] |

| **No Included Containers** |
|---|

⌋

## [ECUC_J1939Tp_00131] Definition of EcucIntegerParamDef J1939TpRxDirectNPduId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxDirectNPduId | | |
| **Parent Container** | J1939TpRxDirectNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

## [ECUC_J1939Tp_00132] Definition of EcucReferenceDef J1939TpRxDirectNPduRef ⌈

| | |
|---|---|
| **Parameter Name** | J1939TpRxDirectNPduRef |
| **Parent Container** | J1939TpRxDirectNPdu |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### 10.1.9 J1939TpRxNSdu

**[ECUC_J1939Tp_00063] Definition of EcucParamConfContainerDef J1939TpRxNSdu** ⌈

| | |
|---|---|
| **Container Name** | J1939TpRxNSdu |
| **Parent Container** | J1939TpRxPg |
| **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. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxNSduId | 1 | [ECUC_J1939Tp_00184] |
| J1939TpRxNSduRef | 1 | [ECUC_J1939Tp_00069] |

| **No Included Containers** |
|---|

⌋

**[ECUC_J1939Tp_00184] Definition of EcucIntegerParamDef J1939TpRxNSduId** ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxNSduId | | |
| **Parent Container** | J1939TpRxNSdu | | |
| **Description** | This is a unique identifier for a received N-SDU. This Id is used in the CancelReceive and ChangeParameter 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** | – | |

▽

△

| | |
|---|---|
| **Scope / Dependency** | scope: ECU<br>withAuto = true |

⌋

### [ECUC_J1939Tp_00069] Definition of EcucReferenceDef J1939TpRxNSduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxNSduRef | | |
| **Parent Container** | J1939TpRxNSdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.10 J1939TpTxFcNPdu

### [ECUC_J1939Tp_00135] Definition of EcucParamConfContainerDef J1939TpTxFcNPdu ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxFcNPdu |
| **Parent Container** | J1939TpRxChannel |
| **Description** | 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.<br>Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxFcNPduTxConfId | 1 | [ECUC_J1939Tp_00168] |
| J1939TpTxFcNPduRef | 1 | [ECUC_J1939Tp_00136] |

**No Included Containers**

⌋

### [ECUC_J1939Tp_00168] Definition of EcucIntegerParamDef J1939TpTxFcNPduTxConfId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxFcNPduTxConfId | | |
| **Parent Container** | J1939TpTxFcNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00136] Definition of EcucReferenceDef J1939TpTxFcNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxFcNPduRef | | |
| **Parent Container** | J1939TpTxFcNPdu | | |
| **Description** | Reference to the Pdu object representing the N-PDU.<br>Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpTxCmNPduRef 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 |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.11 J1939TpTxChannel

### [ECUC_J1939Tp_00059] Definition of EcucParamConfContainerDef J1939TpTxChannel ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxChannel |
| **Parent Container** | J1939TpConfiguration |
| **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. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxCancellationSupport | 0..1 | [ECUC_J1939Tp_00192] |
| J1939TpTxDa | 0..1 | [ECUC_J1939Tp_00180] |
| J1939TpTxDynamicBlockCalculation | 0..1 | [ECUC_J1939Tp_00191] |
| J1939TpTxMaxPacketsPerBlock | 0..1 | [ECUC_J1939Tp_00190] |
| J1939TpTxProtocolType | 0..1 | [ECUC_J1939Tp_00137] |
| J1939TpTxRetrySupport | 0..1 | [ECUC_J1939Tp_00193] |
| J1939TpTxSa | 0..1 | [ECUC_J1939Tp_00181] |
| J1939TpTxChannelComMNetworkHandleRef | 1 | [ECUC_J1939Tp_00195] |

| **Included Containers** | | |
|---|---|---|
| **Container Name** | **Multiplicity** | **Scope / 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.<br>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_00192] Definition of EcucBooleanParamDef J1939TpTxCancellationSupport ⌈

| | |
|---|---|
| **Parameter Name** | J1939TpTxCancellationSupport |
| **Parent Container** | J1939TpTxChannel |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00180] Definition of EcucIntegerParamDef J1939TpTxDa ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxDa | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00191] Definition of EcucBooleanParamDef J1939TpTxDynamicBlockCalculation ⌈

| | |
|---|---|
| **Parameter Name** | J1939TpTxDynamicBlockCalculation |
| **Parent Container** | J1939TpTxChannel |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00190] Definition of EcucIntegerParamDef J1939TpTxMaxPacketsPerBlock ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxMaxPacketsPerBlock | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00137] Definition of EcucEnumerationParamDef J1939TpTxProtocolType ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxProtocolType | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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).<br>This protocol uses two N-PDUs: The CmNPdu and the DtNPdu. | |
| | J1939TP_PROTOCOL_CMDT | J1939 transport protocol type CMDT (Connection Mode Data Transfer).<br>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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00193] Definition of EcucBooleanParamDef J1939TpTxRetrySupport ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxRetrySupport | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00181] Definition of EcucIntegerParamDef J1939TpTxSa ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxSa | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00195] Definition of EcucReferenceDef J1939TpTxChannelComMNetworkHandleRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxChannelComMNetworkHandleRef | | |
| **Parent Container** | J1939TpTxChannel | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### 10.1.12 J1939TpRxFcNPdu

### [ECUC_J1939Tp_00144] Definition of EcucParamConfContainerDef J1939TpRxFcNPdu ⌈

| | |
|---|---|
| **Container Name** | J1939TpRxFcNPdu |
| **Parent Container** | J1939TpTxChannel |
| **Description** | 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.<br>Please note: This sub container is only required when J1939TpTxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpRxFcNPduId | 1 | [ECUC_J1939Tp_00145] |
| J1939TpRxFcNPduRef | 1 | [ECUC_J1939Tp_00146] |

| **No Included Containers** |
|---|

⌋

### [ECUC_J1939Tp_00145] Definition of EcucIntegerParamDef J1939TpRxFcNPduId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxFcNPduId | | |
| **Parent Container** | J1939TpRxFcNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00146] Definition of EcucReferenceDef J1939TpRxFcNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpRxFcNPduRef | | |
| **Parent Container** | J1939TpRxFcNPdu | | |
| **Description** | Reference to the Pdu object representing the N-PDU.<br>Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpRxCmNPdu Ref of the corresponding J1939TpRxChannel. | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### 10.1.13 J1939TpTxCmNPdu

### [ECUC_J1939Tp_00138] Definition of EcucParamConfContainerDef J1939TpTx CmNPdu ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxCmNPdu |
| **Parent Container** | J1939TpTxChannel |
| **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. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxCmNPduTxConfId | 1 | [ECUC_J1939Tp_00170] |
| J1939TpTxCmNPduRef | 1 | [ECUC_J1939Tp_00139] |

**No Included Containers**

⌋

### [ECUC_J1939Tp_00170] Definition of EcucIntegerParamDef J1939TpTxCmNPduTxConfId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxCmNPduTxConfId | | |
| **Parent Container** | J1939TpTxCmNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00139] Definition of EcucReferenceDef J1939TpTxCmNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxCmNPduRef | | |
| **Parent Container** | J1939TpTxCmNPdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.14 J1939TpTxDtNPdu

### [ECUC_J1939Tp_00142] Definition of EcucParamConfContainerDef J1939TpTxDtNPdu ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxDtNPdu |
| **Parent Container** | J1939TpTxChannel |
| **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. |
| **Configuration Parameters** | |

| Included Parameters | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxDtNPduTxConfId | 1 | [ECUC_J1939Tp_00171] |
| J1939TpTxDtNPduRef | 1 | [ECUC_J1939Tp_00143] |

| No Included Containers |
|---|

⌋

### [ECUC_J1939Tp_00171] Definition of EcucIntegerParamDef J1939TpTxDtNPduTxConfId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxDtNPduTxConfId | | |
| **Parent Container** | J1939TpTxDtNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00143] Definition of EcucReferenceDef J1939TpTxDtNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxDtNPduRef | | |
| **Parent Container** | J1939TpTxDtNPdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

### 10.1.15 J1939TpTxPg

#### [ECUC_J1939Tp_00070] Definition of EcucParamConfContainerDef J1939TpTxPg ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxPg |
| **Parent Container** | J1939TpTxChannel |
| **Description** | Parameter group transmitted by the J1939 transport layer. |
| **Configuration Parameters** | |

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxPgDynLength | 1 | [ECUC_J1939Tp_00148] |
| J1939TpTxPgPGN | 1 | [ECUC_J1939Tp_00150] |

| **Included Containers** | | |
|---|---|---|
| **Container Name** | **Multiplicity** | **Scope / 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 that 8 bytes. This N-PDU produces a meta data item of type CAN_ID_32.<br>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_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.<br>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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

### [ECUC_J1939Tp_00150] Definition of EcucIntegerParamDef J1939TpTxPgPGN ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxPgPGN | | |
| **Parent Container** | J1939TpTxPg | | |
| **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** | – | |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.16 J1939TpTxDirectNPdu

### [ECUC_J1939Tp_00140] Definition of EcucParamConfContainerDef J1939TpTxDirectNPdu ⌈

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

| **Included Parameters** | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxDirectNPduTxConfId | 1 | [ECUC_J1939Tp_00169] |
| J1939TpTxDirectNPduRef | 1 | [ECUC_J1939Tp_00141] |

| **No Included Containers** |
|---|

⌋

### [ECUC_J1939Tp_00169] Definition of EcucIntegerParamDef J1939TpTxDirectNPduTxConfId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxDirectNPduTxConfId | | |
| **Parent Container** | J1939TpTxDirectNPdu | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>withAuto = true | | |

⌋

### [ECUC_J1939Tp_00141] Definition of EcucReferenceDef J1939TpTxDirectNPduRef ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxDirectNPduRef | | |
| **Parent Container** | J1939TpTxDirectNPdu | | |
| **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 |
| **Scope / Dependency** | scope: local | | |

⌋

## 10.1.17 J1939TpTxNSdu

### [ECUC_J1939Tp_00147] Definition of EcucParamConfContainerDef J1939TpTxNSdu ⌈

| | |
|---|---|
| **Container Name** | J1939TpTxNSdu |
| **Parent Container** | J1939TpTxPg |
| **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. |
| **Configuration Parameters** | |

| Included Parameters | | |
|---|---|---|
| **Parameter Name** | **Multiplicity** | **ECUC ID** |
| J1939TpTxNSduId | 1 | [ECUC_J1939Tp_00149] |
| J1939TpTxNSduRef | 1 | [ECUC_J1939Tp_00151] |

| No Included Containers |
|---|

⌋

### [ECUC_J1939Tp_00149] Definition of EcucIntegerParamDef J1939TpTxNSduId ⌈

| | | | |
|---|---|---|---|
| **Parameter Name** | J1939TpTxNSduId | | |
| **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** | – | |
| **Scope / Dependency** | scope: ECU<br>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 |
| **Scope / Dependency** | scope: local | | |

⌋

# 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_00416, SRS_BSW_00417, SRS_BSW_00419, SRS_BSW_00423, SRS_BSW_00427, SRS_BSW_00433, SRS_BSW_00437

⌈These requirements are not applicable to this specification.⌋

# 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 R24-11

### B.1.1 Added Specification Items in R24-11

none

### B.1.2 Changed Specification Items in R24-11

[SWS_J1939Tp_00041] [SWS_J1939Tp_00116]

### B.1.3 Deleted Specification Items in R24-11

none