



Elektrobit

# EB tresos<sup>®</sup> ECU Configuration Wizard documentation

product release 8.8.3



Elektrobit Automotive GmbH  
Am Wolfsmantel 46  
91058 Erlangen, Germany  
Phone: +49 9131 7701 0  
Fax: +49 9131 7701 6333  
Email: [info.automotive@elektrobit.com](mailto:info.automotive@elektrobit.com)

## Technical support

<https://www.elektrobit.com/support>

## Legal disclaimer

Confidential information.

ALL RIGHTS RESERVED. No part of this publication may be copied in any form, by photocopy, microfilm, retrieval system, or by any other means now known or hereafter invented without the prior written permission of Elektrobit Automotive GmbH.

All brand names, trademarks, and registered trademarks are property of their rightful owners and are used only for description.

Copyright 2021, Elektrobit Automotive GmbH.

# Table of Contents

1. Overview of EB tresos ECU Configuration Wizard documentation .....	7
2. ECU Configuration Wizard release notes .....	8
2.1. Overview .....	8
2.2. Scope of the release .....	8
2.2.1. Configuration tool .....	8
2.2.2. AUTOSAR modules .....	8
2.2.3. EB (Elektrobit) modules .....	8
2.2.4. MCAL modules and EB tresos AutoCore OS .....	9
2.3. Module release notes .....	9
2.3.1. Configurators module release notes .....	9
2.3.1.1. Change log .....	9
2.3.1.2. New features .....	39
2.3.1.3. EB-specific enhancements .....	39
2.3.1.4. Deviations .....	39
2.3.1.5. Limitations .....	40
2.3.1.6. Open-source software .....	42
3. ECU Configuration Wizard user's guide .....	43
3.1. Overview .....	43
3.2. Background information .....	43
3.3. Common mapping aspects .....	43
3.3.1. Naming rules .....	43
3.3.1.1. Name mangling algorithm for importers .....	43
3.3.1.2. Name prefixing .....	44
3.3.1.3. Instance handling .....	44
3.3.2. Signal offsets .....	46
3.3.3. Rx NM PDUs .....	46
3.3.4. Duplication of Tx N-PDU configuration containers .....	47
3.3.5. Collection of N-PDU elements .....	47
3.3.6. PDU routing .....	47
3.3.7. PDU length calculation .....	48
3.4. Importer parameter mappings for AUTOSAR 4.0 modules .....	48
3.4.1. Overview .....	48
3.4.2. Can .....	48
3.4.2.1. CanObjectId configuration .....	52
3.4.3. CanIf .....	52
3.4.4. CanNm .....	58
3.4.5. CanSM .....	63
3.4.6. CanTp .....	64
3.4.7. CanTSyn .....	68

3.4.8. Com .....	71
3.4.8.1. Configuration for partial networking ERA and EIRA PDUs and signals .....	84
3.4.8.2. Configuration for partial networking IRA signals .....	85
3.4.9. ComM .....	86
3.4.10. Csm .....	90
3.4.10.1. Csm for SecOC .....	91
3.4.10.2. Csm for Tls .....	93
3.4.10.3. Retrieval of curve information .....	98
3.4.10.4. Retrieval of signature scheme information .....	98
3.4.11. Dcm .....	98
3.4.11.1. Configuration of subservices for Diagnostic Session Control .....	125
3.4.11.2. Configuration of subservices for Diagnostic Security Access .....	125
3.4.11.3. Configuration of subservices for Diagnostic Read Data By Identifier .....	126
3.4.11.4. Configuration of subservices for Diagnostic Write Data By Identifier .....	126
3.4.11.5. Configuration of subservices for Diagnostic IOControl .....	126
3.4.11.6. Configuration of subservices for Diagnostic Write Memory By Address .....	127
3.4.11.7. Configuration of subservices for Diagnostic Read Memory By Address .....	127
3.4.11.8. Configuration of subservices for Diagnostic Routine Control .....	127
3.4.11.9. Configuration of subservices for Diagnostic Read Data By Periodic Identifier..	128
3.4.11.10. Configuration of subservices for Diagnostic Dynamically Define Data Identifier .....	128
3.4.11.11. Configuration of subservices for Diagnostic Clear Diagnostic Information .....	129
3.4.11.12. Configuration of subservices for Diagnostic Read DTC Information .....	129
3.4.11.13. Configuration of diagnostic subservices EcuReset .....	130
3.4.11.14. Configuration of diagnostic subservices ComControl .....	130
3.4.11.15. Configuration of subservices for Diagnostic RequestOnBoardMonitoringTestResults .....	131
3.4.11.16. Configuration of subservices for Diagnostic RequestVehicleInfo .....	131
3.4.11.17. Configuration of subservices for Diagnostic RequestDownload .....	132
3.4.11.18. Configuration of subservices for Diagnostic RequestUpload .....	132
3.4.11.19. Configuration of subservices for Diagnostic DataTransfer .....	132
3.4.11.20. Configuration of subservices for Diagnostic RequestTransferExit .....	133
3.4.11.21. Configuration of diagnostic subservices ControlDTCSetting .....	133
3.4.11.22. Determining the access permissions for diagnostic service instances .....	133
3.4.11.23. Determining the type for a routine signal .....	134
3.4.11.24. Configuration of DcmProcessingConditions .....	134
3.4.11.25. Configuration of DcmDspPid .....	134
3.4.11.25.1. Configuration of DcmDspPidData .....	135
3.4.11.25.2. Configuration of DcmDspPidDataSupportInfo .....	135
3.4.11.25.3. Configuration of DcmDspPidService01 .....	135
3.4.11.25.4. Configuration of DcmDspPidSupportInfo .....	136
3.4.12. Dem .....	136

3.4.13. DoIP .....	145
3.4.14. EcuC .....	148
3.4.15. EcuM .....	152
3.4.16. Eth .....	153
3.4.17. EthIf .....	153
3.4.18. EthSM .....	156
3.4.19. EthSwt .....	156
3.4.20. EthTSyn .....	160
3.4.21. FiM .....	165
3.4.22. Fr .....	166
3.4.23. FrArTp .....	169
3.4.24. FrIf .....	172
3.4.25. FrNm .....	177
3.4.26. FrSM .....	182
3.4.27. FrTp .....	182
3.4.28. FrTSyn .....	187
3.4.29. IpduM .....	190
3.4.30. J1939Dcm .....	197
3.4.31. J1939Nm .....	198
3.4.32. J1939Rm .....	199
3.4.33. J1939Tp .....	202
3.4.34. LdCom .....	206
3.4.35. Lin .....	208
3.4.36. LinIf .....	208
3.4.37. LinSM .....	215
3.4.38. LinTp .....	215
3.4.39. Nm .....	217
3.4.40. PduR .....	220
3.4.41. Sd .....	222
3.4.42. SecOC .....	232
3.4.43. SoAd .....	237
3.4.43.1. SoAdSocketConnectionGroup .....	237
3.4.43.2. Parameters derived from the remote SOCKET-ADDRESS .....	240
3.4.43.3. Parameters derived from the local SOCKET-ADDRESS .....	240
3.4.43.4. Local SOCKET-ADDRESS elements configured by Sd .....	241
3.4.43.5. Parameters derived from other entities .....	242
3.4.43.6. SoAdPduRoute and SoAdSocketRoute containers .....	243
3.4.43.7. SoAdPduRoute .....	245
3.4.43.8. SoAdPduRoute/SoAdPduRouteDest .....	245
3.4.43.9. SoAdSocketRoute .....	246
3.4.43.10. SoAdSocketRoute/SoAdSocketRouteDest .....	246
3.4.43.11. SoAdRoutingGroup .....	247

3.4.43.12. PDU types without transmission support on SoAdSocketConnectionGroup level .....	247
3.4.43.13. SocketConnOrSocketConnBundleRef .....	247
3.4.44. SomelpTp .....	248
3.4.45. StbM .....	249
3.4.46. Tcplp .....	251
3.4.47. Tls .....	257
3.4.47.1. TlsCipherSuite Mapping .....	258
3.4.48. UdpNm .....	260
3.4.49. Xcp .....	264
4. Bibliography .....	266



# 1. Overview of EB tresos ECU Configuration Wizard documentation

Welcome to the EB tresos ECU Configuration Wizard (ECU Configuration Wizard) product documentation.

This document provides:

- ▶ [Chapter 2, “ECU Configuration Wizard release notes”](#): release notes for the ECU Configuration Wizard module
- ▶ [Chapter 3, “ECU Configuration Wizard user's guide”](#): containing background information and instructions

## 2. ECU Configuration Wizard release notes

### 2.1. Overview

This chapter provides the ECU Configuration Wizard product specific release notes. General release notes that are applicable to all products are provided in the EB tresos AutoCore Generic documentation. Refer to the general release notes in addition to the product release notes documented here.

### 2.2. Scope of the release

#### 2.2.1. Configuration tool

Your release of EB tresos AutoCore is compatible with the release of the EB tresos Studio configuration tool:

- ▶ EB tresos Studio: 28.1.0 b210701-0227

#### 2.2.2. AUTOSAR modules

The following table lists the AUTOSAR modules that are part of this ECU Configuration Wizard release.

Module name	AUTOSAR version and revision	SWS version and revision	Module version	Supplier
No AUTOSAR modules available				

Table 2.1. Hardware-Independent Modules specified by the AUTOSAR standard

#### 2.2.3. EB (Elektrobit) modules

The following table lists all modules which are part of this release but are not specified by the AUTOSAR standard. These modules include tooling developed by EB or they may hold files shared by all other modules.

Module name	Module version	Supplier
<a href="#">Configurators</a>	2.8.35	Elektrobit Automotive GmbH

Table 2.2. Modules not specified by the AUTOSAR standard



## 2.2.4. MCAL modules and EB tresos AutoCore OS

For information about MCAL modules and OS, refer to the respective documentation, which is available as PDF at `$TRESOS_BASE/doc/3.0_EB_tresos_AutoCore_OS` and `$TRESOS_BASE/doc/5.0_MCAL_modules`<sup>1</sup>. It is also available in the online help in EB tresos Studio. Browse to the folders `EB tresos AutoCore OS` and `MCAL modules`.

## 2.3. Module release notes

### 2.3.1. Configurators module release notes

- ▶ Module version: 2.8.35.B439717
- ▶ Supplier: Elektrobit Automotive GmbH

#### 2.3.1.1. Change log

This chapter lists the changes between different versions.

##### Module version 2.8.35

2021-06-25

- ▶ Extended ComTransformers for Tls and Csm to configure the TLS cipher suites `TLS_ECDHE_ECDSA_WITH_NULL_SHA` and `TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`
- ▶ ASCCONFIGURATORS-1082 Extended DcmTransformer to configure `DcmDspDidRead` and `DcmDspDidWrite` subcontainers if either `DiagnosticServiceInstance` or `DiagnosticServiceClass` references a valid `DiagnosticAccessPermission`

##### Module version 2.8.34

2021-05-28

- ▶ Extended ComTransformer for EthIf to configure `EthIfPortStartupActiveTime`
- ▶ ASCCONFIGURATORS-1215 Fixed known issue: J1939Rm Transformer configures identical short names for `J1939RmUpperComIPdu` and `J1939RmLowerComIPdu`

---

<sup>1</sup>`$TRESOS_BASE` is the location at which you installed EB tresos Studio.

### Module version 2.8.33

2021-04-30

- ▶ Add support for `ConsumedServiceInstances` in multicast `SocketAddress` elements
- ▶ Internal module performance improvement. This module version update does not affect module functionality
- ▶ ASCCONFIGURATORS-1208 Fixed known issue: Com Transformer calculates incorrect `ComSignalType` when `dataTypePolicy` is set to `networkRepresentationFromComSpec`
- ▶ ASCCONFIGURATORS-1205 Fixed known issue: Dcm Transformer uses potentially invalid values to configure `DcmDspReadMemoryRangeInfo` and `DcmDspWriteMemoryRangeInfo` container creation
- ▶ Added Support AUTOSAR `SignalLength` SDG for the configuration of `ComBitSize`
- ▶ ASCCONFIGURATORS-1206 Fixed known issue: Dcm Transformer uses potentially invalid values to configure `DcmDspDidFreezeCurrentState`, `DcmDspDidResetToDefault` and `DcmDspDidShortTermAdjustment`
- ▶ ASCCONFIGURATORS-1207 Fixed known issue: Dcm Transformer uses potentially invalid values to configure `DcmDspVehInfoDataOrder` and `DcmDspVehInfoDataSize`
- ▶ Extended DcmTransformer to configure `DcmDemClientRef` for `DcmDslProtocolRow`

### Module version 2.8.32

2021-03-05

- ▶ Adapted ComTransformer for EthTSyn with fixed `GlobalTimeDomain` and `Port Structure`
- ▶ Extended ComTransformer for Eth to support the configuration according to the AUTOSAR 4.3 schema
- ▶ Extended ComTransformer for SecOC to support the configuration of `SecOCAuthPduHeaderLength`
- ▶ Extended DcmTransformer to configure `DcmDspDataEndianness` for `DcmDspData`
- ▶ ASCCONFIGURATORS-1189 Fixed known issue: Dem Transformer stops with an error when processing `DiagnosticDataElements` with identical names

### Module version 2.8.31

2021-02-12

- ▶ ASCCONFIGURATORS-1178 Fixed known issue: Dem Transformer uses potentially invalid value to configure `DemAgingCycleCounterThreshold`
- ▶ ASCCONFIGURATORS-1177 Fixed known issue: Dem Transformer uses potentially invalid `Id` field to configure `DemDidIdentifier`
- ▶ Extended transformer for SecOC to support the authentication of `DoIP` and `SomeIpTp` Sdus

- ▶ ASCCONFIGURATORS-1176 Fixed known issue: Dcm Transformer uses a potentially invalid ID field for `DcmDspMemoryIdInfo` container creation

#### Module version 2.8.30

2021-01-22

- ▶ Added transformer for `Tls`

#### Module version 2.8.29

2020-12-18

- ▶ Added transformer for `EcuM` to support the configuration of synchronous PNC wake-up
- ▶ Extended transformer for `SoAd` to consider two remote `SocketAddress` elements as distinct if at least one of them has a dynamic port assigned
- ▶ Extended transformer for `TcpIp` to configure the `TcpIpIpSecConfig` container
- ▶ Extended transformers for `TcpIp` and `Sd` to stop accessing the `EthernetCommunicationConnector` to `NetworkEndpoint` references since they will become obsolete in the future

#### Module version 2.8.28

2020-10-23

- ▶ Extended transformers for `Dcm` to support the multiple `DiagnosticServiceTable` and to configure `DcmDspMaxDidToRead` parameter for `DiagnosticReadDataByIdentifier`
- ▶ Added transformers for `J1939Tp`, `J1939Nm`, `J1939Rm`, and `J1939Dcm`
- ▶ Extended transformers for `Dcm` to support Parallel Processing for OBD and UDS

#### Module version 2.8.27

2020-09-25

- ▶ Extended transformers for `SoAd` and to `SoAd` to support the Ethernet service and communication model of the AUTOSAR R19-11 release
- ▶ ASCCONFIGURATORS-1151 Fixed known issue: Transformer for `Sd` does not create all `SdServerService/SdClientService` containers
- ▶ Extended transformer for `EthTSyn` to create one additional `EthTSynPortConfig` container if the configured ECU instance contains a switch

- ▶ Extended transformers for `TcpIp` and `SoAd` to configure `TcpIpIpFragmentationRxEnabled`, `TcpIpIpFragmentationTxEnabled`, and `SoAdSocketUdpRetryEnabled` depending on the MTU that is configured at the `EthernetCommunicationConnector`
- ▶ Extended transformer for `Com` such that no `ComSignal` containers are created for `ISignals` that have a bit offset assigned which does not lie in the user data area of its `NmPdu`

#### Module version 2.8.26

2020-07-31

- ▶ Extended transformer for `FrTp` to support the configuration of a single `FrTpConnection` container using two `FlexrayTpConnection` elements
- ▶ Extended transformer for `Com` to support the configuration of `ComSignalType` for signals of type `ARRAY` that have an `SwBaseType` without valid `length` assigned

#### Module version 2.8.25

2020-06-19

- ▶ Extended Transformer for `Sd` to support the configuration of `SdMulticastEventSoConRef` even if the `SoAdSocketConnectionGroup` container of the `SdEventHandler` contains more than one multicast `SoAdSocketConnection`
- ▶ Extended Transformer for `PduR` to support the simultaneous routing and local reception of `DoIp` SDUs that are received via functional addressing or are related to a `DiagnosticConnection`
- ▶ Added Transformer for the `Csm` module
- ▶ Internal module improvement. This module version update does not affect module functionality
- ▶ Extended transformer for `DoIp` to configure `DoIpChannel` containers for `DoIpTpConnection` elements that do not contain an `ident` sub element
- ▶ ASCCONFIGURATORS-1125 Fixed known issue: `DcmDslProtocolID` is not configured for `DoIpTpConnections`
- ▶ ASCCONFIGURATORS-1124 Fixed known issue: Empty `DemExtendedDataClass` containers are configured

#### Module version 2.8.24

2020-05-22

- ▶ Extended Transformer for `SecOC` to support the configuration of `SecOCMessageLinkLen` and `SecOCMessageLinkPos`

- ▶ Extended Transformers for `EcuC`, `CanIf`, `CanNm`, `SoAd` and `TcpIp` to consider `IPduMapping.pdu-MaxLength` for the calculation of the payload length of a PDU
- ▶ Extended Transformer for `IpduM` to configure an Rx PDU fan-in for contained PDUs if they have the same header-id values assigned and their `ContainerIPdus` are configured as `AcceptAll`
- ▶ Extended Transformer for `TcpIp` to support the configuration of `TcpIpIPv4PathMtuTimeout`, `TcpIpIPv4PathMtuEnabled`, `TcpIpIPv6PathMtuTimeout` and `TcpIpIPv6PathMtuEnabled`
- ▶ ASCCONFIGURATORS-1114 Fixed known issue: During the import of the Diagnostic Extract, the `DiagnosticEvent` and `DiagnosticAging` are in common namespace and the duplicate names are not appended with a suffix "\_1" to the short name

### Module version 2.8.23

2020-04-27

- ▶ Extended Transformers for `FrIf` and `FrArTp` to support merging of `FrArTpConnection` elements with the same local/remote Tp address pair and to support the configuration of one and the same Tx N-PDU in different slots
- ▶ Extended Transformer for `Com` to support the configuration of PDUs that are received via `SecOC` and TP API
- ▶ Extended Transformer for `Com` to support the configuration of `ComDataInvalidAction`

### Module version 2.8.21

2020-02-21

- ▶ Extended Transformer for `FrArTp` to add N-PDUs to `FrArTpChannel` containers only if they are actually sent or received by the Ecu Instance
- ▶ ASCCONFIGURATORS-1087 Fixed known issue: `SoAdPduRoute` containers are not created for Broadcast/Multicast `SocketConnectionGroups`
- ▶ Extended Transformer for `DoIp` to support the configuration of more than one `DoIpUdpConnection` for a given remote tester address
- ▶ Extended Transformer for `Com` to support boolean `SwBaseType` elements of arbitrary bit lengths
- ▶ Extended Transformer for `ComM` to support the configuration of `ComMUser` containers
- ▶ Extended Transformers for `Com`, `LdCom`, `SecOC`, and `SoAd` to configure TP API for PDU Triggerings that are referenced by `EthTpConnection` elements

### Module version 2.8.20

2019-10-17

- ▶ Extended Transformer for `SomeIpTp` to support `SomeIpTpChannel` entities
- ▶ Extended Transformer for `TcpIp` to configure `TcpIpTcpKeepAliveEnabled`, extended Transformer for `SoAd` to configure `SoAdSocketTcpKeepAlive`
- ▶ Extended Transformer for `StbM` to support parameters `StbMOffsetCorrectionAdaptionInterval`, `StbMOffsetCorrectionJumpThreshold`, `StbMRateCorrectionMeasurementDuration` and `StbMRateCorrectionsPerMeasurementDuration`
- ▶ Extended Transformer for `LinIf` to prevent the configuration of `LinFrame` elements that the configured Ecu Instance sends or receives via `LinSlave` communication controllers
- ▶ ASCCONFIGURATORS-1074 Fixed known issue: Transformer for CanSM issues Null pointer exception for `CommunicationConnectors` that do not refer to valid `Controllers`

#### Module version 2.8.19

2019-09-06

- ▶ ASCCONFIGURATORS-1064 Fixed known issue: Transformer for `SoAd` does not create Pdu Routes for dynamically configured local IP addresses

#### Module version 2.8.18

2019-07-12

- ▶ Support array types for the `DcmDspRoutineSignalType`
- ▶ ASCCONFIGURATORS-1055 Fixed known issue: Transformers for `Com`, `ComM` issue a `NullPointerException` if PNC vector is missing

#### Module version 2.8.17

2019-06-14

- ▶ Remove prefix for names of `DcmDspData` elements
- ▶ Extended Transformer for `ComM` to configure `ComMNmVariant` as `NONE` if the configured Ecu Instance does not send or receive PDUs on the `ComMChannel`
- ▶ Extended Transformers for `EcuC`, `CanNm`, `FrNm`, `UdpNm`, `PduR` and `Com` to configure received Nm User Data PDUs even if the signals contained in the Nm PDUs have no signal port assigned
- ▶ Extended Transformer for `SoAd` to configure `GeneralPurposeIPdu` elements of category `Dlt_TP` and `Dlt_IF`
- ▶ Extended Transformer for `SoAd` to configure `SoAdTxPduCollectionSemantics`

- ▶ ASCCONFIGURATORS-1041 Fixed known issue: During the import of the Diagnostic Extract, the creation of a configuration container in Dem throws an exception. This exception is caused by an illegal container short name

#### Module version 2.8.16

2019-05-17

- ▶ Extended Transformer for SoAd to support Service Oriented Communication of System Description .arxml files
- ▶ Extended Transformer for StbM to configure StbMisSystemWideGlobalTimeMaster for abstract Global Time Domains
- ▶ ASCCONFIGURATORS-1035 Fixed known issue: Transformer for LinIf configures LinIfCollisionResolvingRef incorrectly
- ▶ ASCCONFIGURATORS-1038 Fixed known issue: Dcm importer wrongly configures the SessionRefs and SecurityLevelRefs for the ServiceInstance value of AccessPermissionValidity

#### Module version 2.8.15

2019-04-18

- ▶ Extended Transformer for Dem to support the configuration of the DemOBDSupport and the DemOBDSupportKind parameters
- ▶ Adapted Transformer for TcpIp to skip the configuration of TcpIpStaticIpAddressConfig containers if the parent TcpIpLocalAddr container has been created for a multicast address that is configured at runtime by Sd
- ▶ Adapted Transformer documentation for Dcm to correctly reflect the information for DiagnosticComControl (0x28) service
- ▶ Adapted Transformer for Dcm to fill the Data Access Interface by the DEXT importer
- ▶ Adapted Transformer documentation for Dcm to write VARIABLE-SIZE instead of VARIABLE-LENGTH
- ▶ Extended Transformer for CanNm to resolve short name collisions in the case that NmPdu elements are transmitted in different CAN networks using identical CAN Ids or CAN Id ranges
- ▶ Adapted Transformer for Dcm to also take into account MAPPED-FLAT-SWC-SERVICE-DEPENDENCY when configuring the DcmDspRoutineUsePort
- ▶ Extended Transformer for Com to configure ComSignalDirection and ComSignalGroupDirection
- ▶ Extended Transformers to support the reception of one and the same PDU via different networks

#### Module version 2.8.13

2019-03-14

- ▶ ASCCONFIGURATORS-1006 Fixed known issue: Transformer for Xcp erroneously adds non-Xcp PDUs

### Module version 2.8.12

2019-02-15

- ▶ Extended Transformer for `Dcm` to configure the `DcmDspRequestRoutineResultsIn` container inside the `Dsp` container, only if the `REQUESTS` tag exists inside the `DiagnosticExtract`
- ▶ Extended Transformer for `IpduM` to configure `IpduMContainerQueueSize` by using `ContainerIPdu.minimumRxContainerQueueSize` and `ContainerIPdu.minimumTxContainerQueueSize`
- ▶ Extended Transformer for `StbM`, `EthTSyn`, `CanTSyn`, and `FrTSyn` to support configuration via AUTOSAR 4.4 `GlobalTimeDomain` elements
- ▶ Extended Transformer for `IpduM` to configure `IpduMContainerTxTriggerMode` of `IpduMContainerTxPdu` to `IPDUM_DIRECT`, if the `IpduMContainerTxPdu` is itself contained in a `SecuredIPdu`, i.e. a PDU that is processed in `SecOC`
- ▶ Extended Transformer for `ComM` to take the `PncMapping` references to `PhysicalChannel` elements into account for setting up the `ComMPnc` lists
- ▶ Extended Transformer for `ComM` to support the configuration of managed `ComMChannel` containers

### Module version 2.8.11

2019-02-05

- ▶ ASCCONFIGURATORS-996 Fixed known issue: `DiagnosticEventHandler` issues `NullPointerException` on Event that is not mapped onto an operation Cycle
- ▶ Extended Transformer for `Dem` to configure the `DiagnosticDemProvidedDataMapping.DataProvider` with the standardized values
- ▶ Extended Transformer for `IpduM` to configure `IpduMContainedTxPduPriority`

### Module version 2.8.10

2019-01-25

- ▶ Extended configuration of `CanTSynGlobalTimeFollowUpTimeout` in Transformer for `CanTSyn`
- ▶ Extended Transformer for `ComM` to configure `ComMPncEthIfSwitchPortGroupRef`
- ▶ Extended Transformers to allow execution in EB tresos Studio 26 and to support transformation of ASR 4.4 system description files
- ▶ Extended Transformers for `CanIf` and `FrIf` to configure `PduR` as upper layer for routed N-PDUs



- ▶ Extended Transformer for `Dem` to configure the `DemDtcStatusAvailabilityMask` parameter inside the `DemGeneral` container

### Module version 2.8.9

2018-12-13

- ▶ ASCCONFIGURATORS-977 Fixed known issue: Transformer for `EcuC` stops with an error if a sent PDU undergoes a tx fanout before `SecOC/IpduM` and before `<Bus<If` at the same time
- ▶ Extended Transformer for `Dcm` to configure the `DcmDspRoutineFixedLength` parameter of a `DcmDspRoutine`
- ▶ Extended Transformer for `Dem` to configure the `DemEventSignificance` parameter inside the `DemEventClass` container
- ▶ ASCCONFIGURATORS-980 Fixed known issue: Transformer for `SoAd` configures `SomeIpTp` N-PDUs as TP API PDUs instead of as IF API PDUs
- ▶ Extended Transformers for `IpduM`, `SecOC`, and `PduR` to support transmission of cryptographic PDUs in container PDUs and routing of cryptographic PDUs
- ▶ Extended Transformer for `Dem` to configure the `DemEventDestination` parameter inside the `DemEventClass` container
- ▶ Extended Transformer for `IpduM` to configure `IpduMContainerQueueSize`

### Module version 2.8.8

2018-10-26

- ▶ ASCCONFIGURATORS-964 Fixed known issue: Transformer for `EcuC` does not configure Routing-only Applicative TP SDUs
- ▶ Extended Transformers for `Com` and `LdCom` to support the configuration of `Tp` API for PDUs that are sent or received via `SomeIpTp`
- ▶ Extended Transformer for `ComM` to configure `ComMPncComSignal` entries as ERA if they are associated with a `COMM_GATEWAY_TYPE_PASSIVE` `ComMChannel`
- ▶ ASCCONFIGURATORS-971 Fixed known issue: Transformer for `FrTSyn` does not configure `FrTSynGlobalTimeSyncDataIDList` correctly

### Module version 2.8.7

2018-09-28

- ▶ Extended Transformer for `SoAd` to support use of "0.0.0.0" IPV4 addresses as dynamic remote IP addresses

- ▶ Extended Transformer for `IPduM` to support parameter `IpduMUnusedAreasDefault` in `IpduMContainerTxPduS`
- ▶ ASCCONFIGURATORS-957 Fixed known issue: Transformer for `SecOC` does not process cryptographic PDUs

#### Module version 2.8.6

2018-08-24

- ▶ Removed configuration of `EthSwtPortSpeed` from Transformer for `EthSwt`

#### Module version 2.8.5

2018-07-31

- ▶ Extended Transformer for `StbM`: Added support for the configuration of `StbMSynchronizedTimeBaseIdentifier`
- ▶ Extended Transformers to ignore Frames, PDUs, and Signals that are sent or received via `LinSlave` elements
- ▶ Extended Transformer for `LdCom` to support the configuration of a received PDU irrespective of its `minimumDelay`, `transmissionModeTrueTiming` and `transmissionModeCondition`
- ▶ Extended Transformer for `Com` to implement OEM specific fallback for configuration of `ComSignalType`
- ▶ Extended Transformer for `EcuC` to exclude secured PDUs and Pdus contained in a secured PDU from the Meta-Data handling
- ▶ Extended Transformer for `Dcm` to configure `DiagnosticDataTransfer`
- ▶ ASCCONFIGURATORS-951 Fixed known issue: Transformer for `IPduM` creates duplicated Contained Rx PDUs

#### Module version 2.8.4

2018-06-22

- ▶ Removed creation of `IPV4` limited broadcast address entries in Transformer for `TcpIp`
- ▶ Extended Transformer for `PduR` to configure `PduRRoutingGroup` containers
- ▶ Extended Transformer for `SecOC` to support the configuration of distinct Secured and Cryptographic PDU containers
- ▶ Extended Transformer for `EcuC` to configure Meta-Data for Ethernet PDUs that are received or sent by server components

- ▶ Extended Transformer for `EthSwt` to configure modified parameter `EthSwtPortPhysicalLayerType` and new parameter `EthSwtPortMacLayerType` in `EthSwtPort`

### Module version 2.8.3

2018-05-25

- ▶ Extended Transformer for `Com` to configure `ISignalIPduGroup` elements that refer to `NmPdu` elements
- ▶ Extended Transformers for `Com`, `SecOC`, `IpduM`, `PduR`, and `EcuC` to support tx fan-out in between these modules
- ▶ Extended Transformers for `Com` and `SomeIP` to support sending and receiving of `ComIPdu` elements via TP API
- ▶ Extended Transformer for `CanIf` to support `CAN_TSYN` as value for `CanIfRxPduUserRxIndicationUL` and `CanIfTxPduUserTxConfirmationUL`
- ▶ Extended Transformer for `Dcm` to configure custom sub-services for `DiagnosticEcuReset`
- ▶ ASCCONFIGURATORS-922 Fixed known issue: `EthIfCtrlMtu` is configured incorrectly
- ▶ Changed Transformer for `EthIf`: Length of VLAN tag is no longer subtracted from `EthIfCtrlMtu`
- ▶ Extended Transformer for `IPduM` to configure the `offset` and `updateIndicationBitPosition` parameters in `IpduMContained[Rx|Tx]Pdu`, added support for `ContainerIPduHeaderTypeEnum.no-Header`
- ▶ Extended Transformer for `LdCom` to support the configuration of a received PDU irrespective of its `transferProperty`
- ▶ Extended Transformer for `DoIp` to support `IPV6` during configuration of `VehicleAnnouncement` container
- ▶ Extended Transformer for `PduR` to configure `PduRTpThreshold` only if the `PduRRoutingPath` container holds exactly one `PduRDest` subcontainer
- ▶ Extended Transformer for `FrTSyn` to configure `FrTSynGlobalTimeSequenceCounterJumpWidth` in the context of a `FrTSynGlobalTimeSlave` container
- ▶ ASCCONFIGURATORS-921 Fixed known issue: `CanIfRxPduDlc` is configured incorrectly
- ▶ Extended Transformer for `SoAd` to configure `SoAdSocketTcpInitiate` only in `Tcplp` `SoAdSocketConnectionGroup` containers that contain exactly one `SoAdSocketConnection` sub container

### Module version 2.8.2

2018-04-20

- ▶ ASCCONFIGURATORS-904 Fixed known issue: `MultiplexedIPdus` that are sent and received at the same time are configured incorrectly

- ▶ **Extended Transformer for Dcm to configure references** `DcmDspReadMemoryRangeSecurityLevelRef` in `DcmDspReadMemoryRangeInfo` **and** `DcmDspReadMemoryRangeByLabelInfo`, `DcmDspWriteMemoryRangeSecurityLevelRef` in `DcmDspWriteMemoryRangeInfo` **and** `DcmDspWriteMemoryRangeByLabelInfo`
- ▶ **Extended Transformer for Dcm to configure** `DiagnosticRequestOnBoardMonitoringTestResults`
- ▶ **Extended Transformer for Com to configure** `ComIPduGroupRef` **only for** `ComPduGroup` **containers whose direction is not opposite to the referencing PDU**
- ▶ **Extended Transformer for Dcm to configure** `DiagnosticRequestVehicleInfo`

### Module version 2.8.1

2018-03-16

- ▶ **ASCONFIGURATORS-890 Adapted Transformer for Dcm to enable the** `DcmProcessingConditions` **container only in the case when a** `DcmModeCondition` **or a** `DcmModeRule` **exists**
- ▶ **Extended Transformer for Dcm to configure the** `OBDMode_0x0A` (`DiagnosticRequestEmissionRelatedDTCPermanentStatus`) **service**
- ▶ **Extended Transformer for ComM to configure the parameter** `ComMPncPrepareSleepTimer`
- ▶ **Extended Transformers for Nm, UdpNm, CanNm, and FrNm to configure the parameters** `NmRepeatMsgIndEnabled`, `NmNodeDetectionEnabled`, `NmNodeIdEnabled` **either by using the** `NmCluster` **data or by using the** `NmEcu` **data as a fallback**
- ▶ **Extended Transformer for SecOC to configure parameters** `SecOCSecured[Rx|Tx]PduOffset` **and** `SecOCSecured[Rx|Tx]PduLength` **in** `SecOCRxPduSecuredArea`
- ▶ **Extended Transformer for StbM to support the configuration of** `StbMSynchronizedTimeBase` **container for** `GlobalTimeDomains` **that are not linked to any network**
- ▶ **Extended Transformer for SecOC to configure parameter** `SecOCAuthenticationBuildAttempts` **in** `SecOCRxPduProcessing` **and** `SecOCTxPduProcessing`
- ▶ **Extended Transformer for Dcm to support the configuration of** `DcmDspMemory` **container for the** `DiagnosticWriteMemoryByAddress` **and** `DiagnosticReadMemoryByAddress` **services.**
- ▶ **Extended Transformer for ComM to also take routed PDUs into account for setting up the** `PNC` **to** `ComM-Channel` **references**

### Module version 2.8.0

2018-02-16

- ▶ **Extended Transformers for DoIP and PduR to not configure** `DoIP` **SDUs that are either sent and have their** `DiagPduType` **field set to** `DiagRequest` **or that are received and have their** `DiagPduType` **field set to** `DiagResponse`

- ▶ Extended Transformer for `FrTSyn` to configure `FrTSynGlobalTimeOfsDataIDList` and `FrTSynGlobalTimeSyncDataIDList` configuration containers
- ▶ Extended Transformer for `Dem` to configure the `DemInternalDataElementClass` and `DemExternalCSDDataElementClass` choices of the `DemDataElementClass ChoiceContainer`
- ▶ ASCCONFIGURATORS-878 Fix configuration of `DcmDsdSubServiceId` for the `DiagnosticComControl`
- ▶ ASCCONFIGURATORS-879 Fix configuration of Data Access Interface in the case of a `DiagnosticIOControlService`
- ▶ Extended Transformer for `Dcm` to configure `DiagnosticRequestDownload`, `DiagnosticRequestUpload`, `DiagnosticRequestTransferExit`
- ▶ Extended Transformer for `StbM` to configure parameters `StbMTimeLeapFutureThreshold`, `StbMTimeLeapPastThreshold`, and `StbMClearTimeleapCount`
- ▶ Extended Transformer for `SecOC` to configure parameters `SecOCAuthDataFreshnessLen`, `SecOCAuthDataFreshnessStartPosition`, `SecOCUseAuthDataFreshness`, and `SecOCSecuredRxPduVerification`
- ▶ Added Transformer for `SomeIpTp`
- ▶ Transformers for `EcuC`, `CanNm`, `FrNm`, `UdpNm`, `PduR`, `Com`, and `ComM` now configure PNC ERA PDUs and signals for ACTIVE as well as PASSIVE PNC gateways
- ▶ Adapted Transformer for `CanTSyn` to configure the `CanTSynGlobalTimeSequenceCounterJumpWidth` parameter in `CanTSynGlobalTimeSlave`

### Module version 2.7.11

2018-01-19

- ▶ Internal module improvement. This module version update does not affect module functionality

### Module version 2.7.10

2017-12-15

- ▶ Extended Transformer for `Dcm` to support `DcmDspPid` configuration
- ▶ Extended Transformer for `Dcm` to configure the `OBDMode_0x02` (`DiagnosticRequestPowertrainFreezeFrameData`) service
- ▶ Extended Transformer for `Dcm` to configure the `OBDMode_0x01` (`DiagnosticRequestCurrentPowertrainDiagnosticData`) service
- ▶ Extended Transformer for `Dcm` to configure the `OBDMode_0x04` (`DiagnosticClearResetEmissionRelatedInfo`) service
- ▶ Extended `DcmTransformer` for `DcmDspComControl` to configure the `DcmDspComControlSubNode` sub-container

- ▶ Extended Transformer for `SecOC` to configure the `SecOCPduType` parameter for authentic PDUs
- ▶ ASCCONFIGURATORS-859 Fixed known issue: `ComTransferProperty` is configured incorrectly for `SignalGroups`
- ▶ Extended Transformer for `CanNm` to configure the `CanNmNodeIdEnabled` parameter for `NmEcus`
- ▶ ASCCONFIGURATORS-864 Fixed known issue: `ComTimeout` and `ComFirstTimeout` are configured incorrectly for `SignalGroups`

### Module version 2.7.9

2017-11-17

- ▶ Extended Transformer for `EthSwt` to also accept value of `CouplingPort.VlanMembership.defaultPriority` if all `defaultPriority` values of a `CouplingPort` are identical
- ▶ Modified the import of the Variable-size DID signals.
- ▶ Extend the `DcmTransformer` in order to also configure the `DcmDspRequestResultsRoutineSupported` and `DcmDspStopRoutineSupported`
- ▶ Adapted Transformer for `IpduM`: `IpduMContainedRxInContainerPduRef` is only configured if `RxAcceptContainedIPdu` of the associated `ContainerIPDU` is set to `ACCEPT-CONFIGURED`
- ▶ Adapted Transformer for `SoAd`: `SoAdSocketRemoteIpAddress` is now explicitly configured if `Ipv4Configuration/ipv4Address` or `Ipv6Configuration/ipv6Address` is set to `ANY`
- ▶ Extended Transformer for `Dcm` to configure the `OBDMODE_0x08` (`RequestControlOfOnBoardDevice`) service
- ▶ Support fragmentation of PDU size bigger than MTU in `SoAdTransformer` and `TcpIpTransformer`

### Module version 2.7.8

2017-10-20

- ▶ Extended Transformer for `Dcm` to configure the `OBDMODE_0x03` and `OBDMODE_0x07` services
- ▶ Improved handling of `ConsumedServiceInstance` elements linked to `ProvidedServiceInstance` elements which in turn are not connected to any `SocketConnection` or `SocketConnectionBundle`
- ▶ Modified the configuration of `DcmDspDidInfo` and of its subcontainers: `DcmDspDidControl`, `DcmDspDidRead` and `DcmDspDidWrite`

### Module version 2.7.7

2017-09-22

- ▶ Removed workaround in the configuration of `IpduMContainerTxTriggerMode` that was required by a `SecOC` restriction which has been resolved in the meantime
- ▶ Added support for the configuration of `ContainerIPdu` elements that are sent and received by the same `EcuInstance`
- ▶ Extended Transformers for `CanTSyn`, `FrTSyn` and `EthTSyn` to support AUTOSAR RFC 75119
- ▶ Extended Transformer for `Com` to configure the `ComTransferProperty` of an `I-SIGNAL-GROUP` by taking the `ComTransferProperty` values of the contained `I-SIGNAL` elements into account
- ▶ Extended Transformer for `CanTSyn` to configure time domain specific data-ID lists
- ▶ Extended Transformer for `SecOC` to reflect the `SecOC` parameter changes in ACG 8.3 and ACG 8.4
- ▶ Adapted Transformers for `CanNm`, `FrNm`, `UdpNm` and `Nm` to reflect that several parameters in the modules have been moved from the general containers to the `Nm` channel related containers
- ▶ ASCCONFIGURATORS-818 Fixed known issue: Validation error for `DcmDspRoutineInfo` if no signal is configured
- ▶ Extended `DcmTransformer` for `DcmDspComControl` to configure the `DcmDspComControlAllChannel` and `DcmDspComControlSpecificChannel` subcontainers

#### Module version 2.7.6

2017-08-24

- ▶ ASCCONFIGURATORS-808 Fixed known issue: Incomplete import of Diagnostic configuration for `DcmDspDataType`, `DcmDspDataSize`, and `DcmDspRoutineSignalType`
- ▶ ASCCONFIGURATORS-815 Fixed known issue: Received `SecuredIPdus` with `rxSecurityVerification` set to `false` are not routed via Gateway
- ▶ ASCCONFIGURATORS-814 Fixed known issue: Configuration of one and the same PDU for sending and receiving on the same VLAN does not work

#### Module version 2.7.4

2017-07-28

- ▶ Extended Transformer for `PduR` to support the fan-out of PDUs on different VLANs of the same Ethernet network
- ▶ Extended Transformer for `Com` to support the configuration of 64-bit signed/unsigned signal datatypes
- ▶ Modified Transformer to correctly create and configure `DcmDspDidInfo` for `DiagnosticIoControlService`
- ▶ Extended Transformer for `DcmProcessingCondition` to configure `DcmModeRule` and `DcmModeCondition`

- ▶ Extended Transformer for Can to make instance suffix creation robust against invalid `RxIdentifierRanges`
- ▶ ASCCONFIGURATORS-805 Fixed known issue: Received `SecuredIPdu` in `ContainerIPDU` with `rxSecurityVerification` set to `false` causes error during configuration import
- ▶ ASCCONFIGURATORS-807 Fixed known issue: Importer stops with an error if VLANs of Switch Ports are not connected to Ecu Instance
- ▶ Modified `DemTransformer` (`DiagnosticExtendedDataRecordHandler` and `DiagnosticEnableConditionGroupHandler`) to reduce the number of configured `DemExtendedDataClass` and `DemEnableConditionGroup` containers

### Module version 2.7.3

2017-06-29

- ▶ Extended Transformer for ComM to configure `ComMPncComSignalChannelRef` in ERA `ComMPncComSignal` containers
- ▶ Extended Transformer for EthIf to support `MaximumTransmissionUnit` from `EthernetCommunicationConnector` as well as from `EthernetCommunicationController`
- ▶ Extended Transformer for SecOC to support the configuration of `GeneralPurposeIPdu` elements in `SecOCTxAuthenticPduLayer` or `SecOCRxAuthenticPduLayer` containers
- ▶ Extended Transformer for PduR to support the fanout of sent `SecuredIPdu` elements
- ▶ Extended Transformer for EthIf and EthSwt to support managed switching of switch ports

### Module version 2.7.2

2017-06-02

- ▶ Extended Transformer for Dcm to configure `DiagnosticPeriodicID`
- ▶ Extended Transformer for PduR to support the configuration of routing-only multiplexed PDUs and the configuration of multiplexed PDUs that contain routing-only demultiplexed PDUs
- ▶ Extended Transformer for PduR, CanIf, FrIf, and LinIf to support the configuration of routed Xcp PDUs
- ▶ Modified `DiagnosticDIDHandler`, `DiagnosticDynamicallyDefineDataIdentifierHandler`, and `DiagnosticIOControl` to reduce the number of configured `DcmDspDidInfo` containers
- ▶ Extended Transformer for SecOC to support `SecureCommunicationFreshnessProps` and `SecureCommunicationAuthenticationProps` for the configuration of SecOC PDUs
- ▶ Extended Transformer for IpduM to support the configuration of `IpduMContainerTxTriggerMode`
- ▶ Extended Transformers for CanTSyn and FrTSyn to support `GlobalTimeDomain.globalTimePduTriggering` as well as `GlobalTimeDomain.globalTimePdu`



### Module version 2.7.1

2017-05-05

- ▶ Modified DemTransformer to reduce the number of configured DemFreezeFrameClass and DemFreezeFrameRecNumClass elements
- ▶ Configured DemDTCClass to support DTC of type DiagnosticTroubleCodeObd
- ▶ Changed the way how the DcmDspDataType is configured based on Byte Array Signals
- ▶ Extended Transformer for Dcm to configure DcmDslBuffer
- ▶ Improved the configuration of DiagnosticOperationCycles and DiagnosticAgingCycles
- ▶ Extended DcmTransformer to set the DcmDslProtocolSessionRef reference to every DcmDslProtocolRow

### Module version 2.7.0

2017-03-31

- ▶ Extended Transformer for Dcm to configure ClearDiagnosticInformation
- ▶ Extended Transformer for Dcm to configure DiagnosticWriteMemoryByAddress services
- ▶ Extended Transformer for Dcm to configure DiagnosticReadMemoryByAddress services
- ▶ Extended Transformers for CanNm, FrNm, and UdpNm to configure <Bus>NmPnResetTime
- ▶ Extended Transformer for Dem to configure property <Bus>DemAgingCycleRef from the DemEventClass
- ▶ Extended Transformers for UdpNm and Nm to support UdpNmChannels associated with EthernetPhysicalChannels/VLANs
- ▶ Extended Transformer for Com to take ISignal.iSignalType into account for the calculation of signal data types
- ▶ Extended Transformer for LinIf to support the configuration of AssignNad, AssignFrameId, and UnassignFrameId frames using AUTOSAR 4.3.0 LinSlaveConfig entities
- ▶ Extended Transformer for Com to support the configuration of ComSignal/ComFirstTimeout
- ▶ Extended Transformers for CanTSyn and FrTSyn to support the configuration of data-ID lists
- ▶ Extended Transformer for EthTSyn to support the configuration of data-ID lists and EthTSynFramePrio
- ▶ Extended Transformers for Com and SoAd to support the IPV4 and IPV6 address value ANY
- ▶ Extended Transformer for UdpNm to support the configuration of UdpNmImmediateNmCycleTime and UdpNmImmediateNmTransmissions
- ▶ Extended Transformers for CanNm, FrNm, UdpNm, and Nm to use NmCluster.nmPncParticipation for the configuration of <Bus>NmComUserDataSupport and <Bus>NmPnEnabled
- ▶ Extended Dcm to support DcmDslProtocolPriority

- ▶ Added Transformer for `FIM`
- ▶ Set the `DiagnosticPeriodicRate` category in the `DiagnosticPeriodicTransmission` field in the `DcmDsp` container

### Module version 2.6.10

2017-03-03

- ▶ Extended Transformer for `Dcm` to configure `DiagnosticIOControl`
- ▶ Configured `DcmDsdSidTabSubfuncAvail` parameter for `DiagnosticEcuReset`, `DiagnosticRoutineControl` and `DiagnosticReadDTCInformation` services
- ▶ Added Transformer for `Xcp`
- ▶ Modified `Dem` and `Dcm` configurators to not create an additional `configSet` container if one already exists
- ▶ ASCCONFIGURATORS-666 Fixed known issue: The `DcmRteUsage` is configured based on other parameters `DcmDspDataUsePort` and `DcmDspRoutineUsePort`
- ▶ Added support for IPV6 IP addresses to Transformers for `Tcplp` and `SoAd`
- ▶ Moved the available subfunction for diagnostic service `DynamicallyDefineDataIdentifier` from the configuration of the `DataIdentifier` (DID) to the configuration of the diagnostic service
- ▶ ASCCONFIGURATORS-652 Fixed known issue: `Dcm` importer wrongly adds session and security levels for a `DiagnosticService`
- ▶ ASCCONFIGURATORS-719 Fixed known issue: `Dem` configuration generates an error if `DiagnosticDataIdentifier` and `DiagnosticDataElement` have the same `ShortName`

### Module version 2.6.9

2017-02-03

- ▶ ASCCONFIGURATORS-651 Fixed known issue: Fixed `Dcm` error if a DID and one of its aggregated `DiagnosticDataElement` have the same `SHORT-NAME`
- ▶ ASCCONFIGURATORS-662 The `DcmDsdSubServiceSecurityLevelRefs` for the `SubServices` of `DiagnosticSecurityAccess` will not be configured.
- ▶ Extended Transformer for `SecOC` to support the configuration of secured PDUs of type `DcmIPdu`
- ▶ Extended Transformer for `SecOC` to support the configuration of secured PDUs for which `IPduPort.rxSecurityVerification` is set to `false`
- ▶ Extended Transformer for `ComM` to check whether a given PNC ID actually lies within the specified PNC vector
- ▶ Extended Transformer for `Tcplp` to add limited broadcast address entries if they are needed for DHCP

- ▶ Created ECU parameter configuration in the Bsw module Dcm for UDS service `ReadDataByPeriodicIdentifier` from information contained in the diagnostic extract
- ▶ Extended Dcm to configure the `DcmDspDataUsePort` parameter concerning the cases described by `ServiceNeeds/DiagnosticValueNeeds.processingStyle`
- ▶ ASCCONFIGURATORS-665 Fixed known issue: Fixed Dem configurator to configure the `DemEventPriority` parameter
- ▶ Extended Transformer for Dcm to configure `DynamicallyDefineDataIdentifier`

### Module version 2.6.8

2017-01-05

- ▶ ASCCONFIGURATORS-652 Fixed known issue: Fixed Dcm importer wrongly adds session and security levels for a `DiagnosticService`
- ▶ Extended Transformer for Can: Add support for the configuration of `CanObjectId` in multi-CanController scenarios
- ▶ Extended Transformer for EthIf to support `EthIfController`, `EthIfPhysController`, and `EthIfSwitch` configuration containers

### Module version 2.6.7

2016-12-02

- ▶ ASCCONFIGURATORS-654 Fixed known issue: Transformer for DoIP does not configure all `DoIPTcpConnection` containers
- ▶ Extended Transformer for Dcm to configure `ControlDTCSetting`
- ▶ Extended Transformer for IpduM to support multiplexed PDUs and secured PDUs In container PDUs
- ▶ Added Transformer for SecOC

### Module version 2.6.6

2016-11-04

- ▶ Added Transformer for FrArTp
- ▶ Extended Transformers for Com and PduR to support bidirectional routing of PDUs and signals
- ▶ Extended Transformer for SoAd: `ProvidedServiceInstance` and `ConsumedEventGroup` elements are considered for the configuration of `SoAdSocketFramePriority`

### Module version 2.6.5

2016-10-21

- ▶ ASCCONFIGURATORS-606 Fixed known issue: The parameter `DcmDspDataInfoRef` is not set and its container `DcmDspData` is not referenced by DIDs, if no Base Data Type can be determined
- ▶ Extended Transformer for Dcm to configure `DiagnosticEcuReset`
- ▶ ASCCONFIGURATORS-607 Fixed known issue: The parameter `DcmDspRoutineSignalLength` is not configured if the `DiagnosticDataElement` evaluates to a non-array element
- ▶ ASCCONFIGURATORS-608 Fixed known issue: Import ECU configuration erroneously configures `SubServices` for the Dcm `RoutineControl` service
- ▶ Extended Transformer for TcpIp to configure `TcpIpAssignmentPriority`
- ▶ Adapted Transformer for Nm: `NmChannelId` is not configured any more
- ▶ Adapted Transformer for TcpIp to support AUTOSAR 4.2.2 parameter structure
- ▶ Extended Transformers for PduR and SoAd to support PDU routing between `<Bus>Tp` and `SoAd`
- ▶ Extended Transformer for Sd to support AUTOSAR RFC 73286
- ▶ Added Transformer for EthSwt
- ▶ Extended Transformer for Dem to configure `DiagnosticExtendedDataRecord`

#### Module version 2.6.4

2016-09-09

- ▶ ASCCONFIGURATORS-614 Fixed known issue: Transformer for Dcm configures `DcmDslProtocolRxTesterSourceAddr` incorrectly
- ▶ Extended Transformer for DoIP to support configurations without `DiagnosticConnections` in the imported system model
- ▶ Added Transformers for `CanTSyn` and `FrTSyn`
- ▶ Added Transformer for AUTOSAR 4.0 Dem: Added support for configuring `DiagnosticTroubleCode`, `DiagnosticTroubleCodeGroup`, `DiagnosticEvent`, `DiagnosticEnableCondition`, `DiagnosticEnableConditionGroup` and `DiagnosticIndicator` into Dem
- ▶ Extended Transformer for Dem to configure `DiagnosticDataElements` and `DiagnosticFreezeFrames`
- ▶ Extended Transformer for Dcm to configure `DiagnosticReadDTCInformation`
- ▶ Extended Transformer for Dcm to configure `/DcmDspRoutine/DcmDspRoutineUsePort`

#### Module version 2.6.3

2016-08-05

- ▶ ASCCONFIGURATORS-604 Fixed known issue: Configuration of vehicle announcement in DoIP assumes incorrect remote IP address

- ▶ ASCCONFIGURATORS-599 Fixed known issue: Default Buffer Assignment does not support CAN 2.0 and CAN FD PDUs with identical CAN Ids

#### **Module version 2.6.2**

2016-07-01

- ▶ ASCCONFIGURATORS-594 Fixed known issue: Upper layer of `UserDefinedIPdus` and `GeneralPurposeIPdus` is configured incorrectly in `<Bus>If` modules
- ▶ Extended Transformer for `Tcp` to configure at least one `TcpIpAddrAssignment` container for each `TcpIpLocalAddr` entry

#### **Module version 2.6.1**

2016-05-25

- ▶ Extended Transformer for `SoAd` to support `SoAdSocketConnection` elements without `shortLabel` attribute
- ▶ Extended Transformer for `Can`, `Eth`, `Fr`, `Lin` to support the configuration parameter set defined by the AUTOSAR 4.2.2 standard
- ▶ Extended Transformer for `Dcm` to configure `DiagnosticRoutineControl`

#### **Module version 2.6.0**

2016-04-29

- ▶ ASCCONFIGURATORS-569 Fixed known issue: Upper Layer for `UUDT DcmIPdus` is configured incorrectly in `CanIf`, `LinIf`, and `Frlf`

#### **Module version 2.5.5**

2016-04-01

- ▶ ASCCONFIGURATORS-557 Fixed known issue: Creation of the `Dcm` configuration from system description does not calculate the value of `DcmDspDataSize` correctly for non-array data instances
- ▶ Extended Transformers for `Can` and `CanIf` to create dedicated HOHs for CAN-FD Frames

#### **Module version 2.5.4**

2016-03-04

- ▶ ASCCONFIGURATORS-558 Fixed known issue: Transformer for LinIf does not support LinSlaveConfig elements without LinSlaveConfigIdent
- ▶ Extended Transformer for SoAd to configure `SoAdSocketTcpKeepAliveProbesMax`, `SoAdSocketTcpKeepAliveInterval`, and `SoAdSocketTcpKeepAliveTime`
- ▶ Extended Transformer for ComM to configure `ComMPNCGatewayType`

### Module version 2.5.3

2016-02-05

- ▶ Added input data validity checks to Transformer for Dcm

### Module version 2.5.2

2016-01-15

- ▶ ASCCONFIGURATORS-533 Fixed known issue: Transformer for Com configures `ComSignalDataInvalidValue` and `ComSignalInitValue` for `UINT8_N` signals incorrectly
- ▶ ASCCONFIGURATORS-534 Fixed known issue: Transformer for Sd creates unnecessary `SdConsumedMethods` and `SdProvidedMethods` configuration containers
- ▶ ASCCONFIGURATORS-537 Fixed known issue: Transformer for ComM configures `ComMChannelId` incorrectly
- ▶ Extended Transformer for LinIf: Add support for `LinSlaveConfig` elements
- ▶ ASCCONFIGURATORS-506 Fixed known issue: Transformers for PduR and Com do not conform to `TPS_SYST_01056`
- ▶ Extended Transformers: Resolved naming conflicts for PDUs received with the same CAN ID on different busses
- ▶ ASCCONFIGURATORS-535 Fixed known issue: Transformer for CanIf does not support CAN FD PDU configuration according to Autosar 4.2.2

### Module version 2.5.1

2015-11-06

- ▶ ASCCONFIGURATORS-518 Fixed known issue: Configuration import into Com incorrectly reports error on inconsistent endianness in group signals
- ▶ Removed Transformers for `ComXf`, `SomelpXf`, `E2EXf`
- ▶ Adapted Transformer for PduR: Add routing paths for `UserDefinedIPdus` and `GeneralPurposelPdus`
- ▶ ASCCONFIGURATORS-525 Fixed known issue: ComM channel retrieval for Multiplexer PDUs leads to internal assertion failure

- ▶ Adapted Transformer for Dolp: Adaptations according to DolP related changes in AUTOSAR 4.2.2 system model
- ▶ Added Transformer for Dcm

#### **Module version 2.5.0**

2015-10-09

- ▶ Adapted Transformer for CanNm, UdpNm: Nm PDU user data byte length is now calculated by subtracting the non user data byte length from the total PDU byte length, removed support for obsolete parameter UdpNmBusLoadReductionEnabled
- ▶ Adapted Transformer for CanIf: Extended support for CAN Id range reception according to AUTOSAR RFC 66324
- ▶ Added support for Com datatype retrieval for DataTypePolicy "transformingISignal"
- ▶ Adapted Transformer for EthTSyn, StbM: Added support for the configuration of EthTSynGlobalTimeTx-Period, StbMOffsetTimeBase
- ▶ Adapted Transformer for CanNm: Implemented parameter mapping of CanNmCarWakeUpFilterEnabled, nmCarWakeUpRxEnabled according to AUTOSAR RFC 65423

#### **Module version 2.4.4**

2015-08-14

- ▶ Adapted Transformer for TcpIp: TcpIpAddressType is now also configured for TcpIpLocalAddr entries that have TcpIpStaticIpAddressConfig/TcpIpStaticIpAddress set to ANY
- ▶ Adapted Transformer for UdpNm: UdpNmChannelConfig container is only configured for VLANs on which Nm PDUs are sent
- ▶ Adapted Transformer for SoAd: SoAdSocketConnectionGroup/SoAdPduHeaderEnable is set to true if at least one transmitted PDU is associated with a valid header id
- ▶ Added proxy UdpNm Transformer to package dreisoft.tresos.comimporter.api.transformer.asr41 since the extension of the UdpNm module expects the Transformer to reside in that package
- ▶ Adapted Transformer for Sd: SdInstance containers are only created for VLANs in which the configured ECU executes at least one Client Service or Server Service
- ▶ Adapted Transformer for ComXf, SomeIPXf, E2EXf: Transformer technologies that contain either "1" or "1.0.0" as version number are processed now

#### **Module version 2.4.3**

2015-06-19

- ▶ Adapted Transformers for SoAd and Sd to support seamless service migration/AUTOSAR RFC 61738

#### **Module version 2.4.2**

2015-05-22

- ▶ Update Transformer for Nm according to changes in AUTOSAR RFC 61777
- ▶ ASCCONFIGURATORS-467 Fixed known issue: Transformer for Com issues "zero length BigInteger" error message
- ▶ Added Transformers for EthTSyn, StbM
- ▶ Implemented ComDataType retrieval in Transformer for Com according to AUTOSAR RFC 65384
- ▶ ASCCONFIGURATORS-469 Fixed known issue: Transformer for IpduM configures IpduMSelectorField-Position in MostSignificantByteFirst selector fields incorrectly
- ▶ Added Transformer for E2EXf
- ▶ Added Transformer for ComXf
- ▶ Extended Transformers for Com, IPduM, PduR, CanIf, FrIf, SoAd, EcuC to support Container/Contained PDUs

#### **Module version 2.4.1**

2015-02-20

- ▶ Transformer for LinIf now configures LinIfComMNetworkHandleRef
- ▶ Transformer for FrIf now configures FrNm fan-in PDU according to FlexRay slot number, base cycle and cycle repetition parameters
- ▶ Transformers for Com and ComM now configure sent PNC IRA signals
- ▶ Transformers for EcuC, CanNm, FrNm, UdpNm, PduR, Com, ComM now configure PNC ERA PDUs and signals only for ACTIVE PNC gateways
- ▶ Added Transformer for SomelpXf

#### **Module version 2.4.0**

2015-01-07

- ▶ ASCCONFIGURATORS-430 Fixed known issue: Transformer for Com configures initial values and invalid values of UINT8\_N signals incorrectly
- ▶ ASCCONFIGURATORS-440 Fixed known issue: Transformers stop with an error if direct Tp SDU of FrT-pConnection is not available



- ▶ Added support for 1..\* cardinality of CanNmRxPdu containers in Transformer for CanNm
- ▶ Added support for PNC ERA PDU configuration in Transformers for EcuC, CanNm, FrNm, UdpNm, PduR, Com, ComM
- ▶ Added support for PNC Identifier configuration according to AUTOSAR RFC 52483 in Transformer for ComM

### **Module version 2.3.0**

2014-10-03

- ▶ Made Sd Transformer robust against duplicated routing groups
- ▶ ASCCONFIGURATORS-423 Fixed known issue: Transformer for Com generates wrong ComSignalType for signals assigned to "Array" SwBaseType
- ▶ Extended Transformers to support AUTOSAR 4.2.1 system model files as input
- ▶ Added support for dynamic length signals in Com

### **Module version 2.2.2**

2014-09-05

- ▶ Added Transformers for CanSM, FrSM, LinSM and EthSM
- ▶ Added Transformer for DoIP
- ▶ Added Transformer for LdCom

### **Module version 2.2.1**

2014-08-07

- ▶ Adapt configuration of SdClientServiceActivationRef, SdServerServiceActivationRef according to AUTOSAR 4.1.3 Upstream Mapping
- ▶ Adapt Transformer for Eth so that the MAC address is configured according to AUTOSAR
- ▶ Added Transformer for UdpNm
- ▶ Transformer for Sd now configures Sd Control PDUs
- ▶ Transformer for PduR handles routing of Diagnosis PDUs according to AUTOSAR RFC 63555
- ▶ Transformer for Com now supports dataTypePolicy values other than "legacy"
- ▶ ASCCONFIGURATORS-387 Fixed known issue: FrTpTransformer configures FrTpRxPduPoolRef and FrTpTxPduPoolRef incorrectly

- ▶ ASCCONFIGURATORS-393 Fixed known issue: Transformer for CanTp configures identical symbolic names for received N-PDUs used for Data and FC at the same time
- ▶ ASCCONFIGURATORS-394 Fixed known issue: Transformer for FrIf configures incorrect PDU references to EcuC
- ▶ ASCCONFIGURATORS-391 Fixed known issue: PNC ISignalPduGroups are only considered if the configured ECU instance directly references them
- ▶ Transformer for Sd now supports the configuration of combined Tcp/Udp services
- ▶ ASCCONFIGURATORS-399 Fixed known issue: Transformer for ComM sets ComMNmVariant to "NONE" for LIN Clusters
- ▶ ASCCONFIGURATORS-395 Fixed known issue: Transformers create User Data PDU containers for Nm PDUs containing signals that are not processed by the ECU
- ▶ Transformers for CanNm, FrNm, and UdpNm now configure the links to the associated ComM channels
- ▶ Transformers for ComM now configures ComMChannelId
- ▶ ASCCONFIGURATORS-396 Fixed known issue: Transformer for Nm does not configure NmComUserDataSupport
- ▶ Transformer for Com now configures ComGWMMapping entries for group signals

#### Module version 2.2.0

2014-04-25

- ▶ ASCCONFIGURATORS-349 Fixed known issue: Transformer for PduR sets up incorrect routing paths for reversed FlexRay Tp SDUs
- ▶ ASCCONFIGURATORS-355 Fixed known issue: Transformer for Com does not create gateway mapping entries for signal groups
- ▶ ASCCONFIGURATORS-346 Fixed known issue: Transformer for LinIf configures AssignNad frame incorrectly
- ▶ ASCCONFIGURATORS-354 Fixed known issue: Transformer for LinIf reports an error if it encounters AssignFrameIdRange schedule table entries without PID
- ▶ Extended Transformers to support AUTOSAR 4.1.2 system model files as input; added Transformer for Sd

#### Module version 2.1.10

2014-01-17

- ▶ Integrated handling for CanTp N-PDUs assigned to multiple CanTp connections
- ▶ Integrated AUTOSAR 4.0 ComM Transformer

### Module version 2.1.9

2013-10-11

- ▶ Internal module improvement. This module version update does not affect module functionality

### Module version 2.1.8

2013-09-13

- ▶ Removed configuration of PN ERA PDUs which are not yet supported by CanNm and FrNm

### Module version 2.1.7

2013-06-14

- ▶ ASCCONFIGURATORS-296 Fixed known issue: The configuration of one and the same CanNm PDU for sending and receiving results in a naming conflict
- ▶ Added support for `CONTAINED-I-SIGNAL-I-PDU-GROUP-REF` to detect PNC-enabled NmClusters
- ▶ Added support for routed-only Tp-SDUs in PduR
- ▶ Added configuration of `CanNmPnFilterMaskByte` in CanNm
- ▶ Added support for configuration of `ComTxModeTrue` for sent PDUs without PDU Timing in Com

### Module version 2.1.6

2013-05-10

- ▶ ASCCONFIGURATORS-277 Fixed known issue: Com Transformer restricts signals to be contained at most once per FlexRay frame
- ▶ ASCCONFIGURATORS-279 Fixed known issue: CanNm Transformer issues exception if handed over Can Network contains Rx Nm PDUs but no CAN Frame Triggerings
- ▶ ASCCONFIGURATORS-276 Fixed known issue: If a PDU is sent and received at the same time and fan-out takes place for the sent PDU, the received PDU instance is configured incorrectly
- ▶ ASCCONFIGURATORS-283 Fixed known issue: LinTp Transformer produces name clashes when importing from AUTOSAR 3.x LinTp configurations
- ▶ ASCCONFIGURATORS-286 Fixed known issue: CanTp Transformer cannot handle `CAN-TP-CONNECTIONS` without `TP-SDU-REF`

### Module version 2.1.5

2013-02-08

- ▶ ASCCONFIGURATORS-230 Fixed known issue: Transformer for AUTOSAR 4.0 Com configures signal invalid values and signal init values  $> 2^{53}-1$  incorrectly
- ▶ ASCCONFIGURATORS-250 Fixed known issue: `ComTxModeNumberOfRepetition` is not configured correctly for AUTOSAR 4.0.3 Com configurations
- ▶ ASCCONFIGURATORS-254 Fixed known issue: LinIf Transformer uses wrong set of PDUs for retrieval of TxPdu fan-out information
- ▶ ASCCONFIGURATORS-256 Fixed known issue: PduR Transformer cannot handle <any bus>-PDU to Ethernet-PDU gateway mappings
- ▶ ASCCONFIGURATORS-258 Fixed known issue: FrIf Transformer does not configure `FrIfFluster.gdBit` according to upstream mapping
- ▶ Implemented support for TP-SDUs referenced by multiple PDU triggerings
- ▶ Removed configuration of `<Net>NmPnFilterMaskByte` due to unclear upstream mapping rule
- ▶ Added pure gateway routing support

#### Module version 2.1.4

2012-10-12

- ▶ AUTOSAR 4.0 FrIf/FrNm: Added support for FrNm Rx PDU fan-in
- ▶ ASCCONFIGURATORS-230 Fixed known issue: Transformer for Com does not create `GroupSignal` if `SignalTriggering` is missing

#### Module version 2.1.3

2012-09-14

- ▶ AUTOSAR 4.0 Com: Added support for Com Rx Signal DataFilters
- ▶ AUTOSAR 4.0 Com: Adapted `ComSignalType` configuration: Special handling of type `BOOLEAN`

#### Module version 2.1.2

2012-08-17

- ▶ AUTOSAR 4.0 Com: User-defined prefix is added to group signal containers
- ▶ AUTOSAR 4.0 SoAd: Adapted according to configuration changes
- ▶ AUTOSAR 4.0 Nm: Adapted according to configuration changes
- ▶ AUTOSAR 4.0 Com: Adapted configuration of parameter `ComSignalType` according to AUTOSAR 4.0 Rev 3 System Template

#### **Module version 2.1.1**

2012-06-15

- ▶ AUTOSAR 4.0 SoAd: Implemented configuration of `SoAdSocketConnectionGroup` containers
- ▶ AUTOSAR 4.0 Transformers: PDU Router fan-out support has been added

#### **Module version 2.1.0**

2012-05-16

- ▶ AUTOSAR 4.0 CanIf, CanTp: Adapted parameter configuration according to configuration changes in AUTOSAR 4.0 Rev 3

#### **Module version 2.0.6**

2012-04-13

- ▶ AUTOSAR 4.0 SoAd: Adaptation due to removed/obsolete parameter `SoAdPduHeaderEnable`

#### **Module version 2.0.5**

2012-03-27

- ▶ Added support for the configuration of partial networks for AUTOSAR 4.0 FrNm
- ▶ Integrated AUTOSAR 4.0 Transformers for modules SoAd, EthIf, Eth, TcpIp
- ▶ ASCCONFIGURATORS-187 Fixed known issue: AUTOSAR 4.0 Com: Fixed signal data type calculation algorithm

#### **Module version 2.0.4**

2012-02-17

- ▶ Removed obsolete AUTOSAR 3.x Transformers
- ▶ Added support for the configuration of partial networks for AUTOSAR 4.0 CanNm

#### **Module version 2.0.3**

2012-01-20

- ▶ Integrated AUTOSAR 4.0 Transformers for modules Lin, LinIf, and LinTp

- ▶ Added User Data Nm PDU support to the involved AUTOSAR 4.0 Transformers EcuC, PduR, Com, and CanNm, FrNm

#### **Module version 2.0.2**

2011-10-12

- ▶ Integrated AUTOSAR 4.0 Transformers for modules Fr, FrIf, CanTp, CanNm, FrTp, FrNm, Nm
- ▶ ASCCONFIGURATORS-138 Fixed known issue: FrNm: Fixed Rx/Tx Nm PDU handling

#### **Module version 2.0.0**

2011-09-02

- ▶ ASCCONFIGURATORS-86 Fixed known issue: IpduM: Transformer for AUTOSAR 3.x IpduM configures `IPduMTxSelectorValue`
- ▶ ASCCONFIGURATORS-85 Fixed known issue: IpduM, EcuC: Containers for Demultiplexed PDUs are only created if the Com module actually processes them
- ▶ Integrated AUTOSAR 4.0 Transformers for modules EcuC, Can, CanIf, IpduM, PduR, and Com

#### **Module version 1.1.2**

2011-04-08

- ▶ ASCCONFIGURATORS-51 Fixed known issue: LinIf: Fixed references to EcuC PDU collection
- ▶ Improvement CanTp: Support configuration of multicast CanTp connection channels

#### **Module version 1.1.1**

2011-03-11

- ▶ Improvement CanTp, CanIf, EcuC: Duplication of CanTp Tx N-PDUs has been introduced
- ▶ Improvement LinIf: Transformer configures `LinIfFramePriority` if provided via LDF importer
- ▶ Transformer for generic Nm was added
- ▶ Improvement Com: Support for zero bitsize signals has been added

#### **Module version 1.1.0**

2011-02-03

- ▶ Requirements tracing: Resolve unmapped tests, unmapped requirements
- ▶ Improvement CanNm, FrNm, CanIf, LinIf, Com: Transformers use new Meta-Model 6/AUTOSAR 3.1.4 System parameters for module configuration
- ▶ Improvement CanIf, Can: Transformers use better naming schema for configuring HOH containers

#### **Module version 1.0.2**

2010-11-18

- ▶ ASCCONFIGURATORS-28 Fixed known issue: AUTOSAR 2.1/3.x Com: Com data types calculated correctly from integral system model data types with open ranges

#### **Module version 1.0.1**

2010-10-08

- ▶ Improvement AUTOSAR 3.x LinIf: Conditional frames obtain `LinIfInternalPdu` as `LinIfPduDirection` during com imports
- ▶ ASCCONFIGURATORS-22 Fixed known issue: AUTOSAR 2.1/3.x Com: Bit offsets and length parameters of signal groups are correctly exported

#### **Module version 1.0.0**

2010-09-10

- ▶ First implementation of Configurators

### **2.3.1.2. New features**

- ▶ Configuration support for Csm has been added.

### **2.3.1.3. EB-specific enhancements**

This module is not part of the AUTOSAR specification.

### **2.3.1.4. Deviations**

This module is not part of the AUTOSAR specification.

### 2.3.1.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- ▶ `ComSignalInitValue/ComSignalDataInvalidValue` when importing from FIBEX

**Description:**

When importing from FIBEX, initial and invalid values that have been defined as bit pattern that represent `FLOAT` values are interpreted as integral values and written to `ComSignalInitValue/ComSignalDataInvalidValue` as such.

**Rationale:**

Initial and invalid values are defined as `INTERNAL-CONSTRS` in FIBEX, which requires these values to be integral values.

- ▶ `CanTpChannel/CanTpChannelMode` when importing from FIBEX

**Description:**

When importing from FIBEX, only elements of `CanTpChannel` that contain a single `CanTpRxNSdu/CanTpTxNSdu` are created. Also, the `CanTpChannelMode` of these elements `CanTpChannel` is always set to `CANTP_MODE_HALF_DUPLEX`.

**Rationale:**

FIBEX 3.x only supports a single `TP-CONNECTION` per `TP-CHANNEL`.

- ▶ `ISignal.dataTypePolicy:networkRepresentationFromComSpec` requires either a `SenderReceiverToSignalMapping` or a `SenderReceiverToSignalGroupMapping`

**Description:**

An `ISignal` that has `dataTypePolicy` set to `networkRepresentationFromComSpec` requires a `SenderReceiverToSignalMapping` or a `SenderReceiverToSignalGroupMapping`. These mapping elements reference a `PortPrototype` which contains a `PortComSpec` that defines the `NetworkRepresentation`. A `DataMapping` other than `SenderReceiverToSignalMapping` or `SenderReceiverToSignalGroupMapping` is not supported.

- ▶ `ISignal.dataTypePolicy: Data type retrieval via networkRepresentationFromComSpec` and `ImplementationDataType` is only supported for plain `ISignal` elements

**Description:**

In the AUTOSAR System Template 4.2 Rev 2, [TPS\_SYST\_02006] and [TPS\_SYST\_02079] describe the data type retrieval via `ImplementationDataType` in the case that no `SenderComSpec` or `Receiver-`



`ComSpec` is available. This is currently only implemented for plain signals, not for `ISignals` that represent group signals.

- ▶ `ComIPdu/ComIPduCounter` is not configured

Description:

`ComIPdu/ComIPduCounter` configuration containers are not configured even if the imported AUTOSAR System Description file contains `SIGNAL-I-PDU-COUNTER` elements.

Rationale:

The ACG Com module does not support `ComIPdu/ComIPduCounter` configuration containers.

- ▶ `ComIPdu/ComIPduReplication` is not configured

Description:

`ComIPdu/ComIPduReplication` configuration containers are not configured even if the imported AUTOSAR System Description file contains `SIGNAL-I-PDU-REPLICATION` elements.

Rationale:

The ACG Com module does not support `ComIPdu/ComIPduReplication` configuration containers.

- ▶ `ComSignal/ComRxDataTimeoutAction` and `ComSignalGroup/ComRxDataTimeoutAction` are not configured

Description:

`ComSignal/ComRxDataTimeoutAction` and `ComSignalGroup/ComRxDataTimeoutAction` are not configured even if the imported system model contains `HANDLE-TIMEOUT-TYPE` elements.

Rationale:

The imported system model can contain two or more `HANDLE-TIMEOUT-TYPE` elements that are related to one and the same `ComSignal/ComRxDataTimeoutAction` or `ComSignalGroup/ComRxDataTimeoutAction` parameter. Since these `HANDLE-TIMEOUT-TYPE` elements may contain different values, a configuration is not possible.

- ▶ `FrController/FrFiFo` is not configured

Description:

`FrController/FrFiFo` configuration containers are not configured even if the imported AUTOSAR System Description file contains `FLEXRAY-FIFO-CONFIGURATION` elements.

Rationale:

Not all types of FlexRay communication controllers can be configured using the parameters provided in `FrController/FrFiFo`. It is therefore required to configure `FrController/FrFiFo` by hand, while taking into account the type of FlexRay communication controller that is in use.

- ▶ No support for tx fan-out of PDUs that are contained in `CONTAINER-I-PDU` elements

Description:

While the AUTOSAR System Template 4.2 Rev 1 indicates that PDUs contained in `CONTAINER-I-PDU` elements can be subject to a tx fan-out in the PduR module, the Transformer for PduR currently does not support this feature.

Rationale:

The tx fan-out support for contained PDUs requires a dedicated extension of the Transformer for PduR which has not yet been implemented.

- ▶ `AssignNAD` frames referencing Lin slave nodes which are not declared in the imported LDF file lead to import errors

Description:

During LDF imports, it is required that all `AssignNAD` frames in the LDF file reference Lin slave nodes which are also present in the file. If this is not the case, an error is issued.

Rationale:

The AUTOSAR System Model, which serves as internal data storage during the import, does not provide the possibility to store an `AssignNAD` frame without referencing the related slave node at the same time. Since the EB tresos Studio importer framework is built on top of the AUTOSAR System Model, this parameter cannot be configured.

### 2.3.1.6. Open-source software

Configurators does not use open-source software.

## 3. ECU Configuration Wizard user's guide

### 3.1. Overview

A configuration import consists of two steps. In the first step, one or more configuration files representing a system model - or an ECU extract thereof - are imported into the EB tresos Studio system model representation. In the second step the user selects one of the ECUs of the imported model for configuring a number of AUTOSAR modules in the EB tresos Studio project. Subsequently, the ECU relevant aspects of the imported system model are translated into AUTOSAR module configurations.

This part of the documentation describes the second importer step in more detail. For a description of the first importer step, see the EB tresos Studio user's guide.

- ▶ [Section 3.3, “Common mapping aspects”](#) describes general information that applies to the mapping between system model and AUTOSAR module configurations.
- ▶ [Section 3.4, “Importer parameter mappings for AUTOSAR 4.0 modules”](#) describes the mappings that are made between the system model and the respective AUTOSAR module configurations.

### 3.2. Background information

The chapter *Introduction* and the subchapter *Scope* in [\[3\]](#) provide the background information on the configuration of AUTOSAR modules out of a system model.

### 3.3. Common mapping aspects

This section contains some general information that applies to the mapping between system model and AUTOSAR module configurations.

#### 3.3.1. Naming rules

##### 3.3.1.1. Name mangling algorithm for importers

In AUTOSAR configuration files container names must be valid C-Identifiers and must have a length of at most 32 characters. The names of the containers must be unique within the context of the parent containers. In

order to ensure that all containers created during an import comply to the conditions above, object names are mangled.

Any characters violating the rules for valid C-Identifiers are replaced by underscores (`_`). In case a name exceeds 30 characters it is cut off and an underscore followed by a unique index is added so that the resulting name has a length of at most 30 characters and is unique. Since AUTOSAR allows a length of at most 32 characters, the importers can add prefixes which are at most two characters long without violating AUTOSAR naming rules, e.g. `SG` is added to normal signals, `NW` is added to network signals.

Examples:

- ▶ `TestName9012345678901234567890Foo` becomes `TestName90123456789012345678_1`
- ▶ `Test-Name9012345678901234567890Bar` becomes `TestName90123456789012345678_2`

#### NOTE



#### Automatic modification of the container names

If the check for the maximum container name length is disabled, the container names are mangled to 110 characters instead of 30. For further information on how to check the maximum container name length, see the EB tresos Studio documentation.

To avoid that your container name is modified automatically, enable the check for the maximum container name length. For further information on how to check the maximum container name length, see the EB tresos Studio documentation.

### 3.3.1.2. Name prefixing

Prior to import a prefix can be defined which is added to the `ShortName` of every imported container. The prefix menu line may be left empty if you do not wish a prefix.

The entered prefix is referred to as `<PREFIX>` in this document.

### 3.3.1.3. Instance handling

In AUTOSAR module configurations, instances of frames, PDUs, and signals are handled in different ways, depending on the network. Whereas for CAN and LIN configurations there is usually a one-to-one relationship between prototype and instance, this is usually not the case for FlexRay configurations:

FlexRay frames can be transmitted in multiple communication slots over the bus. One PDU may reside in more than one FlexRay frame. The `FlexRay Interface (FrIf)` handles this one-to-many relationship by always passing the same PDU to the upper layer (e.g. to the `PDU Router, PduR`) even if the PDU is received in different FlexRay communication slots and/or in different frames.

If an upper layer issues a transmit request for a PDU, the `FrIf` sends it in the next FlexRay communication slot that is assigned to a FlexRay frame, which contains the requested PDU.

FlexRay N-PDUs, i.e. PDUs that are processed by the `FlexRay Transport Layer (FrTp)`, must be handled differently by the `FrIf`. If a FlexRay N-PDU is transmitted in multiple frames and/or in multiple FlexRay communication slots, the `FrIf` processes each of the resulting PDU instances as an individual PDU.

If one and the same PDU is received on a given CAN network via two or more different CAN-IDs, the `CanIf` is set up to receive the PDU via a CAN-ID range that contains all the CAN-IDs via which the PDU is received. Since the `CanIf` does not support *standard* and *extended* CAN-IDs in one and the same CAN-ID range, an error is issued if such a configuration is imported. However, the range reception described here does not apply to N-PDUs or NM-PDUs.

Since every instance container of a configuration needs its unique name, and unique names are usually only provided for prototypes, the problem arises how to obtain unique container names for instances of the same prototype. EB tresos Studio provides an option by which the instance names are generated by appending a suffix to the name of the prototype. This guarantees that the instance name is unique.

FlexRay frames obtain different instance suffixes depending on whether they contain FlexRay N-PDUs or not. FlexRay N-PDUs obtain different suffixes than other FlexRay PDUs do. In the following text this is described in detail:

For FlexRay frames that contain N-PDUs, the following information is added to instances of frames:

- ▶ FlexRay identifier (1..2047)
- ▶ Channel information (A|B)
- ▶ Base cycle (0..63)
- ▶ Cycle repetition (1|2|4|8|16|32|64)
- ▶ Transmission direction information (T|R)

An instance of a frame *TestFrame*, received in slot 123, base cycle 10, cycle repetition 32 on channel A would therefore be called `TestFrame_123A1032R`.

If a FlexRay frame does not contain any N-PDU, only the transmission direction information is appended as suffix, e.g. The *TestFrame* above would be called `TestFrame_R`.

For CAN frames, the following information is added to instances of frames:

- ▶ CAN identifier (0..2<sup>29</sup>-1)
- ▶ Transmission direction information (T|R)

An instance of a frame *TestFrame*, received with CAN-ID 123 would therefore be called `TestFrame_123R`.

For LIN frames, the following information is added to instances of frames:

- ▶ LIN identifier (0..63)
- ▶ Transmission direction information (T|R)

Although a LIN frame/PDU can be sent or received in multiple schedule tables or even multiple times in the same schedule, the layer above only sees one instance of this frame/PDU. Therefore the schedule table and the entry position information has been left out in the instance suffix. An instance of a frame `TestFrame`, received with LIN-ID 23 would therefore be called `TestFrame_23R`.

The naming scheme for frame instances can also be applied to PDU instances. The instance of PDU `TestPdu`, which is sent in the frame instance `TestFrame_123R`, is named `TestPdu_123R`.

The naming scheme for signal instances adopts the naming scheme from the I-PDU instance in which it is contained. A signal instance of `TestSignal` transmitted in the PDU instance `TestPdu_123R` is called `TestSignal_123R`.

For an import run, add instance suffixes by checking the **Instance suffix for Frames, PDUs and Signals** checkbox.

If you import FlexRay networks, check the option checkbox to ensure that configuration containers obtain unique names. Since there is usually a one-to-one relationship between prototype and instance in CAN and LIN networks, an instance suffix is not required to obtain unique configuration container names. Uncheck the option checkbox if you import CAN and/or LIN networks to obtain short configuration container names.

Also note that the length of the container names grows considerably if the option is turned on. This usually makes it necessary to turn off the check for the maximum container name length. For instructions on how this check is turned off, see the EB tresos Studio user's guide.

The suffix is referred to by `<INSTSUFFIX>` in this document.

### 3.3.2. Signal offsets

Signals can be packed into PDUs either in big-endian or little-endian format. The endianness information, together with the signal offset parameter defines the bitset within the PDU which is occupied by a signal. Whereas the endianness of a signal within its enclosing PDU is an agreed upon notion, virtually every document format developed its own peculiar definition of the notion signal offset making an implicit offset conversion during import runs necessary.

For information on how the signal offset is defined, see the external document format specification and the AUTOSAR System Template specification.

### 3.3.3. Rx NM PDUs

If Rx NM PDUs are used, only the PDU which has the lowest CAN-ID is considered (per cluster) by the AUTOSAR 3.0/3.1 modules `CanIf`, `EcuC`, and `CanNm`.

### 3.3.4. Duplication of Tx N-PDU configuration containers

If the same outgoing N-PDU is used simultaneously by two `CanTp` connections, i.e. as data N-PDU by an outgoing connection and as flow-control N-PDU by an incoming connection, two configuration containers are created for this N-PDU in the AUTOSAR 3.0/3.1 modules `CanIf`, `EcuC`, and `CanTp`.

The name of the second N-PDU container is suffixed with `_D` and corresponds to the data N-PDU which is used by the outgoing `CanTp` connection.

### 3.3.5. Collection of N-PDU elements

For `CanTp`, `LinTp`, and `FrTp` connections, there is a one-to-one relationship between an N-PDU and the frame in which the N-PDU is sent or received. For that reason an N-PDU is considered sent or received by an ECU if its frame is sent or received. This is determined by the presence of a Tx or Rx frame port that links the frame to the ECU.

An exception to that are `FrArTp` connections. In the context of `FrArTp`, an N-PDU can be contained in a frame that contains other PDUs as well. As a consequence, the presence of an Tx or Rx frame port is not sufficient to determine whether an N-PDU is sent or received by an ECU. In the context of `FrArTp`, Tx or Rx PDU ports linking the N-PDUs to the ECU are required as well.

### 3.3.6. PDU routing

An ECU routes a source PDU to one or more target PDUs if there is a gateway mapping which defines the routing paths from the source PDU to the target PDUs.

In this case, the source PDU is configured as an Rx PDU, and the target PDUs are configured as Tx PDUs in the modules `CanIf`, `FrIf`, `LinIf`, `SoAd`, `CanTp`, `FrTp`, `LinTp`, `DoIP`, `SecOC`, `IpduM`, and `EcuC`. The actual routing from source to destination PDUs is configured in the `PduR` module.

A PDU that is configured as a target PDU in a gateway mapping is not configured in any upper layer BSW module of the `PduR`. A sent `ISignalIPdu`, for instance, is not configured in `Com` if it is configured as a target PDU in a gateway mapping. The reason for that is that the `PduR` only allows one single source for the transmission of a PDU. The source can either be a local transmission request of any upper layer BSW module, e.g. `Com`, or it can be a received source PDU, the content of which is forwarded using the target PDU.

If there is a gateway mapping for a PDU that is related to a *cryptographic* PDU as described in [Section 3.4.42, “SecOC”](#), this gateway mapping either applies to the payload PDU or to the authenticated PDU, i.e. the PDU which `SecOC` exchanges with its upper layer module. If there is a gateway mapping for the *cryptographic* PDU as well, the payload PDU is configured to be routed. If there is no gateway mapping for the *cryptographic* PDU, the authenticated PDU is routed.

### 3.3.7. PDU length calculation

For the calculation of the payload length of a PDU in bytes, two fields are taken into account. The `LENGTH` field of the PDU itself is always used. The second field, `I-PDU-MAPPING/PDU-MAX-LENGTH`, is only taken into account if the `I-PDU-MAPPING` references via `SOURCE-I-PDU-REF` or via `TARGET-I-PDU-REF` a `PDU-TRIGGERING` which in turn references the PDU.

If both `LENGTH` and `I-PDU-MAPPING/PDU-MAX-LENGTH` need to be taken into account for the calculation of the payload length of a PDU, the payload length is considered to be the maximum of both values.

## 3.4. Importer parameter mappings for AUTOSAR 4.0 modules

### 3.4.1. Overview

The following chapters explain how system model parameters are mapped into AUTOSAR modules. For more in-depth information on how to invoke the importers for the different network communication formats, see the EB tresos Studio user's guide.

#### ECU Extract to AUTOSAR parameter mapping

The **System Description Importer** of EB tresos Studio creates configurations for each of the following AUTOSAR modules.

For detailed configuration parameter descriptions, see the EB tresos AutoCore module references.

### 3.4.2. Can

Configuration parameters	Mapping description
CanConfigSet/Can-Controller	<p>For every <code>CAN-COMMUNICATION-CONTROLLER</code> that is connected to a <code>CAN-CLUSTER</code> or <code>J-1939-CLUSTER</code> and to the imported <code>ECU-INSTANCE</code>, a <code>Can-Controller</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>CAN-COMMUNICATION-CONTROLLER</code>.</p> <p><code>CanControllerDefaultBaudrate</code> references the <code>CanControllerBaudrateConfig</code> container created for the <code>CAN-COMMUNICATION-CONTROLLER</code>.</p> <p><code>CanControllerId</code> is configured in such a way that each <code>CanController</code></p>



Configuration parameters	Mapping description
	container obtains a unique <code>CanControllerId</code> value and the set of all <code>CanControllerId</code> values is zero-based and dense.
<code>CanConfigSet/CanController/CanControllerBaudrateConfig</code>	<p><code>CanControllerBaudRate</code> is set to <code>BAUDRATE / 1000</code>, as defined in the <code>CAN-CLUSTER</code> or <code>J-1939-CLUSTER</code> of the <code>CAN-COMMUNICATION-CONTROLLER</code>. If <code>BAUDRATE</code> is not available, <code>CanControllerBaudRate</code> is set to <code>SPEED</code>.</p> <p>The following parameters are set as defined in the <code>CAN-CONTROLLER-CONFIGURATION</code> of the <code>CAN-COMMUNICATION-CONTROLLER</code>:</p> <p><code>CanControllerPropSeg</code> is set to <code>PROP-SEG</code>.</p> <p><code>CanControllerSeg1</code> is set to <code>TIME-SEG-1</code>.</p> <p><code>CanControllerSeg2</code> is set to <code>TIME-SEG-2</code>.</p> <p><code>CanControllerSyncJumpWidth</code> is set to <code>SYNC-JUMP-WIDTH</code>.</p>
<code>CanConfigSet/CanController/CanControllerFdBaudrateConfig</code>	<p>The container <code>CanControllerFdBaudrateConfig</code> is only created if <code>CAN-FD-BAUDRATE</code> contains a valid value. <code>CanControllerFdBaudRate</code> is set to <code>CAN-FD-BAUDRATE / 1000</code>, as defined in the <code>CAN-CLUSTER</code> or <code>J-1939-CLUSTER</code> of the <code>CAN-COMMUNICATION-CONTROLLER</code>.</p> <p>The following parameters are set as defined in the <code>CAN-CONTROLLER-FD-ATTRIBUTES</code> of the <code>CAN-COMMUNICATION-CONTROLLER</code>:</p> <p><code>CanControllerPropSeg</code> is set to <code>PROP-SEG</code>.</p> <p><code>CanControllerSeg1</code> is set to <code>TIME-SEG-1</code>.</p> <p><code>CanControllerSeg2</code> is set to <code>TIME-SEG-2</code>.</p> <p><code>CanControllerSyncJumpWidth</code> is set to <code>SYNC-JUMP-WIDTH</code>.</p> <p><code>CanControllerTrcvDelayCompensationOffset [ns]</code> is set to <code>TRCV-DELAY-COMPENSATION-OFFSET [s]</code>.</p> <p><code>CanControllerTxBitRateSwitch</code> is set to <code>TX-BIT-RATE-SWITCH</code>.</p>
<code>CanConfigSet/CanController/CanFilterMask</code>	<p><code>CanFilterMask</code> containers are only created for <code>Can</code> modules that conform to the AUTOSAR 4.0.3 standard.</p> <p>Two <code>CanFilterMask</code> containers are created for each <code>CanController</code>, named <code>AcceptAllStd</code> and <code>AcceptAllExt</code>. <code>CanFilterMaskValue</code> of theses containers is set to zero per default.</p>

Configuration parameters	Mapping description
	<p>If a parameter tag named <code>canfiltermask.dontcarebit</code> exists in the module's comtransformer extension point, which has its <code>value</code> attribute set to 1, <code>CanFilterMaskValue</code> is set to 0x7ff (AcceptAllStd) and 0xffffffff (AcceptAllExt) instead of zero.</p> <p>If a parameter tag named <code>canfiltermask.std.shiftleft</code> is defined in the module's comtransformer extension point, <code>CanFilterMaskValue</code> of <code>AcceptAllStd</code> is shifted left by the value defined in the <code>value</code> attribute.</p> <p>If a parameter tag named <code>canfiltermask.ext.shiftleft</code> is defined in the module's comtransformer extension point, <code>CanFilterMaskValue</code> of <code>AcceptAllExt</code> is shifted left by the value defined in the <code>value</code> attribute.</p>
CanConfigSet/Can-HardwareObject	<p>If <b>Buffer Assignment in Can/CanIf</b> was set to <b>Create default buffer assignment</b> during the import, <code>CanHardwareObject</code> containers are created for each CAN-COMMUNICATION-CONTROLLER connected to a CAN-CLUSTER or J-1939-CLUSTER of the imported ECU-INSTANCE. The container name depends on CAN-FRAME-RX-BEHAVIOR/CAN-FRAME-TX-BEHAVIOR and on CAN-ADDRESSING-MODE of the FRAME-TRIGGERING associated with the PDU which the ECU-INSTANCE sends or receives. The naming schema for every combination is displayed in <a href="#">Table 3.1, “Rx HOH container names depending on CAN-FRAME-RX-BEHAVIOR and CAN-ADDRESSING-MODE”</a> and <a href="#">Table 3.2, “Tx HOH container names depending on CAN-FRAME-TX-BEHAVIOR and CAN-ADDRESSING-MODE”</a>.</p> <p><code>CanIdType</code> is set depending on the CAN-ADDRESSING-MODE of the FRAME-TRIGGERING elements the HOH containers have been created for. If CAN-ADDRESSING-MODE is set to STANDARD or &lt;undefined&gt;, <code>CanIdType</code> is set to STANDARD. If CAN-ADDRESSING-MODE is set to EXTENDED, <code>CanIdType</code> is set to EXTENDED.</p> <p><code>CanObjectType</code> is set depending on whether the ECU-INSTANCE sends or receives the FRAME-TRIGGERING elements for which the HOH containers were created. If FRAME-TRIGGERING elements are received, <code>CanObjectType</code> is set to RECEIVE. If FRAME-TRIGGERING elements are sent, <code>CanObjectType</code> is set to TRANSMIT.</p> <p><code>CanControllerRef</code> references the <code>CanController</code> container created for the CAN-COMMUNICATION-CONTROLLER of the HOH.</p> <p>The <code>CanFilterMaskRef</code> reference parameter is only configured for Can modules that conform to the AUTOSAR 4.0.3 standard. It references a <code>CanFilterMask</code> container depending on the CAN-ADDRESSING-MODE of the FRAME-</p>

Configuration parameters	Mapping description
	<p>TRIGGERING for which the HOH containers have been created. HOH containers created for CAN-ADDRESSING-MODE STANDARD/&lt;undefined&gt; reference <code>AcceptAllStd</code>, HOH containers created for CAN-ADDRESSING-MODE EXTENDED reference <code>AcceptAllExt</code>.</p> <p><code>CanObjectId</code> is configured according to <a href="#">Section 3.4.2.1, “CanObjectId configuration”</a>.</p> <p><code>CanHandleType</code> is set to BASIC for all HOHs.</p> <p><code>CanFdPaddingValue</code> is only configured for Can modules that conform to the AUTOSAR 4.2.2 standard. The value is set to CAN-COMMUNICATION-CONTROLLER/CAN-COMMUNICATION-CONTROLLER-VARIANTS/CAN-COMMUNICATION-CONTROLLER-CONDITIONAL/CAN-CONTROLLER-ATTRIBUTES/CAN-CONTROLLER-CONFIGURATION/CAN-CONTROLLER-FD-ATTRIBUTES/PADDING-VALUE or, if this parameter is not present, to CAN-COMMUNICATION-CONTROLLER/CAN-COMMUNICATION-CONTROLLER-VARIANTS/CAN-COMMUNICATION-CONTROLLER-CONDITIONAL/CAN-CONTROLLER-ATTRIBUTES/CAN-CONTROLLER-CONFIGURATION/CAN-CONTROLLER-FD-REQUIREMENTS/PADDING-VALUE.</p>
<code>CanConfigSet/Can-HardwareObject/Can-HwFilter</code>	<p>The <code>CanHwFilter</code> container is only created for Can modules that conform to the AUTOSAR 4.2.2 standard.</p> <p><code>CanHwFilterCode</code> is configured to 0. <code>CanHwFilterMask</code> is configured to the filter mask value that allows that all CAN-IDs are received. The description of <code>CanConfigSet/CanController/CanFilterMask</code> in <a href="#">Section 3.4.2, “Can”</a> provides the calculation formulas of these filter mask values.</p>

CAN-ADDRESSING-MODE	STANDARD or <undefined>	EXTENDED
CAN-FRAME-RX-BEHAVIOR		
CAN-20 or <undefined>	<code>HOH_0_&lt;controller name&gt;</code>	<code>HOH_1_&lt;controller name&gt;</code>
CAN-FD	<code>HOH_0_&lt;controller name&gt;</code>	<code>HOH_1_&lt;controller name&gt;</code>
CAN-ANY	<code>HOH_0_&lt;controller name&gt;</code>	<code>HOH_1_&lt;controller name&gt;</code>

Table 3.1. Rx HOH container names depending on CAN-FRAME-RX-BEHAVIOR and CAN-ADDRESSING-MODE

CAN-ADDRESSING-MODE	STANDARD or <undefined>	EXTENDED
CAN-FRAME-TX-BEHAVIOR		
CAN-20 or <undefined>	<code>HOH_2_&lt;controller name&gt;</code>	<code>HOH_3_&lt;controller name&gt;</code>

CAN-FD	HOH_4_<controller name>	HOH_5_<controller name>
--------	-------------------------	-------------------------

Table 3.2. Tx HOH container names depending on CAN-FRAME-TX-BEHAVIOR and CAN-ADDRESSING-MODE

### 3.4.2.1. CanObjectId configuration

The `CanObjectId` handle ID parameters of all `CanHardwareObject` containers are configured in such a way that each `CanHardwareObject` obtains a unique `CanObjectId` value and the set of all values is zero-based and dense. Moreover, the values are distributed in increasing order to the following:

- ▶ RECEIVE `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to 0.
- ▶ RECEIVE `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to 1.
- ▶ ...
- ▶ RECEIVE `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to  $\langle n - 1 \rangle$ .
- ▶ TRANSMIT `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to 0.
- ▶ TRANSMIT `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to 1.
- ▶ ...
- ▶ TRANSMIT `CanHardwareObject` containers that reference the `CanCommunicationController` which has its `CanControllerId` set to  $\langle n - 1 \rangle$ .

### 3.4.3. CanIf

Configuration parameters	Mapping description
<code>CanIfCtrlDrvCfg</code>	<p>If no <code>CanIfCtrlDrvCfg</code> container exists, a new one named <code>&lt;PREFIX&gt;_CanIfCtrlDrvCfg</code> is created. Otherwise the first existing container is used and the following parameters are set:</p> <p><code>CanIfCtrlDrvNameRef</code> references the <code>CanGeneral</code> container in the Can driver module configuration.</p> <p><code>CanIfCtrlDrvInitHohConfigRef</code> references the <code>CanIfInitHohCfg</code> container created during this import.</p>

Configuration parameters	Mapping description
CanIfCtrl- DrvCfg/CanIfCtrlCfg	<p>For every CAN-COMMUNICATION-CONTROLLER connected to a CAN-CLUSTER or J-1939-CLUSTER of the imported ECU-INSTANCE, a CanIfCtrlCfg container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the CAN-COMMUNICATION-CONTROLLER.</p> <p>CanIfCtrlCanCtrlRef references the corresponding CanController container in the Can driver module configuration.</p>
CanIfInitCfg	<p>If no CanIfInitCfg container exists, a new one named CanIfInitCfg is created. Otherwise the first existing container is used and the following sub containers are added:</p>
CanIfInitCfg/ CanIfInitHohCfg	<p>CanIfInitRefCfgSet references the CanConfigSet container in the Can driver module configuration.</p>
CanIfInitCfg/ CanIfInitHoh- hCfg/CanIfHrhCfg	<p>If <b>Buffer Assignment in Can/CanIf</b> has been set to <b>Create default buffer assignment</b> during the import, CanIfHrhCfg containers are created for each CAN-COMMUNICATION-CONTROLLER connected to a CAN-CLUSTER or J-1939-CLUSTER of the imported ECU-INSTANCE. The container name depends on the CAN-FRAME-RX-BEHAVIOR and on the CAN-ADDRESSING-MODE of the FRAME-TRIGGERING associated with the PDU which the ECU-INSTANCE receives. The naming schema for every combination is displayed in <a href="#">Table 3.1, “Rx HOH container names depending on CAN-FRAME-RX-BEHAVIOR and CAN-ADDRESSING-MODE”</a>.</p> <p>CanIfHrhCanCtrlIdRef references the CanIfCtrlCfg container created for the CAN-COMMUNICATION-CONTROLLER of the HOH.</p> <p>CanIfHrhIdSymRef references the corresponding CanHardwareObject container in the Can driver module configuration.</p>
CanIfInitCfg/ CanIfInitHoh- hCfg/CanIfHthCfg	<p>If <b>Buffer Assignment in Can/CanIf</b> has been set to <b>Create default buffer assignment</b> during the import, CanIfHthCfg containers are created for each CAN-COMMUNICATION-CONTROLLER connected to a CAN-CLUSTER or J-1939-CLUSTER of the imported ECU-INSTANCE. The container name depends on the CAN-FRAME-TX-BEHAVIOR and on the CAN-ADDRESSING-MODE of the FRAME-TRIGGERING associated with the PDU which the ECU-INSTANCE sends. The naming schema for every combination is displayed in <a href="#">Table 3.2, “Tx HOH container names depending on CAN-FRAME-TX-BEHAVIOR and CAN-ADDRESSING-MODE”</a>.</p> <p>CanIfHthCanCtrlIdRef references the CanIfCtrlCfg container created for the CAN-COMMUNICATION-CONTROLLER of the HOH.</p>

Configuration parameters	Mapping description
	<p><code>CanIfHthIdSymRef</code> references the corresponding <code>CanHardwareObject</code> container in the Can driver module configuration.</p>
<p><code>CanIfInitCfg/</code> <code>CanIfRxPduCfg</code></p>	<p>For every PDU received or routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) by the imported ECU-INSTANCE a <code>CanIfRxPduCfg</code> container is created.</p> <p>The container name is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the PDU.</p> <p>The following kinds of PDUs are excluded from <code>CanIfRxPduCfg</code> container creation:</p> <ul style="list-style-type: none"> <li>▶ PDUs that are referenced in DYNAMIC-PARTS or STATIC-PARTS of MULTIPLEXED-I-PDU elements.</li> <li>▶ PDUs that are received within CONTAINER-I-PDU elements.</li> </ul> <p>If multiple NM-PDU elements are received via one CAN-CLUSTER or J-1939-CLUSTER, a <code>CanIfRxPduCfg</code> container is only created for the PDU with the lowest CAN-ID (defined either as IDENTIFIER or via RX-IDENTIFIER-RANGE). The range of CAN-IDs of all these NM-PDU elements is stored under <code>CanIfRxPduCanIdRange</code>.</p> <p>If the FRAME-TRIGGERING to which the PDU belongs defines an RX-IDENTIFIER-RANGE or the PDU is received via different FRAME-TRIGGERING elements yielding two or more CAN-ID values, a <code>CanIfRxPduCanIdRange</code> container is created as well.</p> <p><code>CanIfRxPduCanId</code> is set to the CAN-ID of the PDU for which this <code>CanIfRxPduCfg</code> container has been created, if no <code>CanIfRxPduCanIdRange</code> container has been created in the previous step. Otherwise this parameter is not configured.</p> <p><code>CanIfRxPduCanIdMask</code> is set to the RX-MASK value of the CAN-FRAME-TRIGGERING that receives the CAN-FRAME which contains the PDU.</p> <p><code>CanIfRxPduCanIdType</code> is set depending on the values of CAN-ADDRESSING-MODE and CAN-FRAME-RX-BEHAVIOR. If CAN-FRAME-RX-BEHAVIOR is not available, CAN-20 is assumed as default value:</p> <ul style="list-style-type: none"> <li>▶ STANDARD_NO_FD_CAN for STANDARD and CAN-20</li> <li>▶ EXTENDED_NO_FD_CAN for EXTENDED and CAN-20</li> <li>▶ STANDARD_FD_CAN for STANDARD and CAN-FD</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ EXTENDED_FD_CAN for EXTENDED and CAN-FD</li> <li>▶ STANDARD_CAN for STANDARD and ANY</li> <li>▶ EXTENDED_CAN for EXTENDED and ANY</li> </ul> <p>CanIfRxPduHrhIdRef references the CanIfHrhCfg container created for CAN-FRAME-TRIGGERING elements with these CAN-ADDRESSING-MODE and CAN-FRAME-RX-BEHAVIOR values (see <a href="#">Section 3.4.2, “Can”</a>, CanHardware-Object).</p> <p>If CAN-ADDRESSING-MODE is undefined, CanIfRxPduHrhIdRef is not configured either.</p> <p>CanIfRxPduRef references the corresponding container in the EcuC module configuration.</p> <p>CanIfRxPduUserRxIndicationUL is set depending on the type of PDU:</p> <ul style="list-style-type: none"> <li>▶ CAN_NM for NM-PDU elements that are received in CAN-CLUSTER elements.</li> <li>▶ J1939NM for NM-PDU elements that are received in J-1939-CLUSTER elements.</li> <li>▶ CAN_TP for N-PDU elements that are received in CAN-CLUSTER elements, and not routed via gateway.</li> <li>▶ J1939TP for N-PDU elements that are received in J-1939-CLUSTER elements, and not routed via gateway.</li> <li>▶ CAN_TSYN for GENERAL-PURPOSE-PDU elements with its CATEGORY parameter set to GLOBAL_TIME.</li> <li>▶ PDUR for I-SIGNAL-PDU elements, MULTIPLEXED-I-PDU elements, CONTAINER-I-PDU elements, GENERAL-PURPOSE-I-PDU elements, DCM-I-PDU elements, for USER-DEFINED-I-PDU elements, which either have a CATEGORY other than XCP or are routed, and for N-PDU elements that are routed (see <a href="#">Section 3.3.6, “PDU routing”</a>).</li> <li>▶ XCP for XCP-PDU elements.</li> <li>▶ CDD for received USER-DEFINED-I-PDU elements that have their CATEGORY set to XCP and for all other PDU types.</li> </ul>
CanIfInitCfg/ CanIfRxPduCfg/CanIfRxPdu-CanIdRange	CanIfRxPduCanIdRangeLowerCanId is set to the CAN-ID of the PDU for which this CanIfRxPduCfg container has been created, or to LOWER-CAN-ID of the RX-IDENTIFIER-RANGE.

Configuration parameters	Mapping description
	<p><b>CanIfRxPduCanIdRangeUpperCanId</b> is set to the highest CAN-ID of all NM-PDU elements received via the CAN-CLUSTER or J-1939-CLUSTER, or to UPPER-CAN-ID of the RX-IDENTIFIER-RANGE.</p> <p>For <b>CanIfRxPduCanIdRange</b> elements created for received NM-PDU elements a consistency check is performed:</p> <p>In case a CAN-NM-NODE belonging to the imported ECU-INSTANCE exists, which defines a different range via CAN-NM-RANGE-CONFIG, this range overrides the range determined via the NM-PDU elements. If this range is narrower than the range as determined via the NM-PDU elements, a warning is reported.</p>
CanIfInitCfg/ CanIfTxPduCfg	<p>For every PDU sent or routed (see <a href="#">Section 3.3.6, "PDU routing"</a>) by the imported ECU-INSTANCE a <b>CanIfTxPduCfg</b> container is created.</p> <p>The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>The following kinds of PDUs are excluded from <b>CanIfTxPduCfg</b> container creation:</p> <ul style="list-style-type: none"> <li>▶ PDUs that are referenced in DYNAMIC-PARTS or STATIC-PARTS of MULTIPLEXED-I-PDU elements.</li> <li>▶ PDUs that are sent within CONTAINER-I-PDU elements.</li> </ul> <p>If a sent N-PDU is referenced by two CAN-TP-CONNECTION elements, a second <b>CanIfTxPduCfg</b> container is created. The Tx CAN-TP-CONNECTION must reference this N-PDU via DATA-PDU-REF, the Rx CAN-TP-CONNECTION via FLOW-CONTROL-PDU-REF. The direction of the CAN-TP-CONNECTION is determined by the PDU referenced by TP-SDU-REF. The container name of the second <b>CanIfTxPduCfg</b> container is &lt;PREFIX&gt;&lt;name&gt;_D&lt;INSTSUFFIX&gt;.</p> <p><b>CanIfTxPduDlc</b> is configured as described in <a href="#">Section 3.3.7, "PDU length calculation"</a>.</p> <p><b>CanIfTxPduCanId</b> is set to the CAN-ID of the PDU for which this <b>CanIfTxPduCfg</b> container has been created.</p> <p><b>CanIfTxPduCanIdMask</b> is set to the TX-MASK value of the CAN-FRAME-TRIGGERING that sends the CAN-FRAME which contains the PDU.</p> <p><b>CanIfTxPduCanIdType</b> is set depending on the values of CAN-ADDRESSING-MODE and CAN-FRAME-TX-BEHAVIOR. If CAN-FRAME-TX-BEHAVIOR is not available, CAN-20 is assumed as default value:</p>



Configuration parameters	Mapping description
	<p><code>CanIfTxPduBufferRef</code> references the <code>CanIfBufferCfg</code> container created for CAN-FRAME-TRIGGERING elements with these CAN-ADDRESSING-MODE and CAN-FRAME-TX-BEHAVIOR values.</p> <ul style="list-style-type: none"> <li>▶ <code>STANDARD_CAN</code> for STANDARD and CAN-20</li> <li>▶ <code>EXTENDED_CAN</code> for EXTENDED and CAN-20</li> <li>▶ <code>STANDARD_FD_CAN</code> for STANDARD and CAN-FD</li> <li>▶ <code>EXTENDED_FD_CAN</code> for EXTENDED and CAN-FD</li> </ul> <p><code>CanIfTxPduHthIdRef</code> references the <code>CanIfHthCfg</code> container created for CAN-FRAME-TRIGGERING elements with these CAN-ADDRESSING-MODE and CAN-FRAME-TX-BEHAVIOR values (see <a href="#">Section 3.4.2, “Can”</a>, <code>CanHardwareObject</code>).</p> <p>If CAN-ADDRESSING-MODE is undefined, <code>CanIfTxPduHthIdRef</code> is not configured either.</p> <p><code>CanIfTxPduRef</code> references the corresponding container in the <code>EcuC</code> module configuration.</p> <p><code>CanIfTxPduUserTxConfirmationUL</code> is set depending on the type of PDU:</p> <ul style="list-style-type: none"> <li>▶ <code>CAN_NM</code> for NM-PDU elements that are sent in CAN-CLUSTER elements.</li> <li>▶ <code>J1939NM</code> for NM-PDU elements that are sent in J-1939-CLUSTER elements.</li> <li>▶ <code>CAN_TP</code> for N-PDU elements that are sent in CAN-CLUSTER elements, and not routed via gateway.</li> <li>▶ <code>J1939TP</code> for N-PDU elements that are sent in J-1939-CLUSTER elements, and not routed via gateway.</li> <li>▶ <code>CAN_TSYN</code> for GENERAL-PURPOSE-PDU elements with its <code>CATEGORY</code> parameter set to <code>GLOBAL_TIME</code>.</li> <li>▶ <code>PDUR</code> for I-SIGNAL-PDU elements, MULTIPLEXED-I-PDU elements, CONTAINER-I-PDU elements, GENERAL-PURPOSE-I-PDU elements, DCM-I-PDU elements, for USER-DEFINED-I-PDU elements, which either have a <code>CATEGORY</code> other than <code>XCP</code> or are routed, and for N-PDU elements that are routed. For more information, see <a href="#">Section 3.3.6, “PDU routing”</a>.</li> <li>▶ <code>CDD</code> for sent USER-DEFINED-I-PDU elements that have their <code>CATEGORY</code> set to <code>XCP</code> and for all other PDU types.</li> </ul>

Configuration parameters	Mapping description
CanIfInitCfg/CanIf-BufferCfg	<p>If <b>Buffer Assignment in Can/CanIf</b> was set to <b>Create default buffer assignment</b> during the import, CanIfBufferCfg containers are created for each CAN-COMMUNICATION-CONTROLLER connected to a CAN-CLUSTER or J-1939-CLUSTER of the imported ECU-INSTANCE. The container name depends on the CAN-FRAME-TX-BEHAVIOR and on the CAN-ADDRESSING-MODE of the FRAME-TRIGGERING associated with the PDU which the ECU-INSTANCE sends. The naming schema for every combination is displayed in <a href="#">Table 3.2, “Tx HOH container names depending on CAN-FRAME-TX-BEHAVIOR and CAN-ADDRESSING-MODE”</a>.</p> <p>CanIfBufferHthRef references the CanIfInitHohCfg/CanIfHthCfg container created for the CAN-FRAME-TRIGGERING elements with these CAN-ADDRESSING-MODE and CAN-FRAME-TX-BEHAVIOR values.</p>

### 3.4.4. CanNm

Configuration parameters	Mapping description
CanNmGlobalConfig	<p>CanNmComUserDataSupport is set to true if CanNmRxUserDataPduRef or CanNmTxUserDataPduRef is set for any CanNmChannelConfig, or if any CAN-NM-CLUSTER linked to the imported ECU-INSTANCE has its NM-PNC-PARTICIPATION either not defined or set to true. Otherwise CanNmComUserDataSupport is set to false.</p> <p>CanNmPnResetTime is set to PN-RESET-TIME of the configured ECU-INSTANCE.</p> <p>The following parameters are set using the first NM-ECU of the imported ECU-INSTANCE:</p> <p>CanNmUserDataEnabled is set to NM-USER-DATA-ENABLED.</p> <p>CanNmRemoteSleepIndEnabled is set to NM-REMOTE-SLEEP-IND-ENABLED.</p> <p>CanNmBusSynchronizationEnabled is set to NM-BUS-SYNCHRONIZATION-ENABLED.</p> <p>CanNmStateChangeIndEnabled is set to NM-STATE-CHANGE-IND-ENABLED.</p>

Configuration parameters	Mapping description
	<p>CanNmPassiveModeEnabled is set to NM-PASSIVE-MODE-ENABLED. If this parameter is not available, CanNmPassiveModeEnabled is set depending on NM-PASSIVE-MODE-ENABLED of all NM-NODES belonging to the imported ECU-INSTANCE:</p> <ul style="list-style-type: none"> <li>▶ false if at least one NM-NODE has NM-PASSIVE-MODE-ENABLED set to false.</li> <li>▶ true if all NM-NODE elements have NM-PASSIVE-MODE-ENABLED set to true.</li> </ul> <p>Otherwise CanNmPassiveModeEnabled is not set.</p> <p>CanNmPduRxIndicationEnabled is set to NM-PDU-RX-INDICATION-ENABLED.</p> <p>CanNmComControlEnabled is set to NM-COM-CONTROL-ENABLED.</p> <p>CanNmMainFunctionPeriod is set to NM-CYCLETIME-MAIN-FUNCTION.</p> <p>If inconsistencies are detected among parameters of multiple NM-ECU elements, a warning is reported.</p> <p>The following parameters are set using the first CAN-NM-CLUSTER-COUPLING of all CAN-NM-CLUSTER elements connected to the imported ECU-INSTANCE:</p> <p>CanNmBusLoadReductionEnabled is set to NM-BUS-LOAD-REDUCTION-ENABLED.</p> <p>CanNmImmediateRestartEnabled is set to NM-IMMEDIATE-RESTART-ENABLED.</p> <p>If inconsistencies are detected among parameters of multiple CAN-NM-CLUSTER-COUPLING elements, a warning is reported.</p> <p>If a CAN-NM-CLUSTER configured as partial networking cluster (PNC) (see <a href="#">Section 3.4.14, "EcuC"</a>) belongs to the imported ECU-INSTANCE, the following parameters are set:</p> <p>CanNmPnEiraRxNSduRef references the corresponding container in the EcuC module configuration.</p> <p>CanNmPnEiraCalcEnabled is set to true.</p>

Configuration parameters	Mapping description
	<p>CanNmPnInfo/CanNmPnInfoOffset is set to PNC-VECTOR-OFFSET. If PNC-VECTOR-OFFSET is not defined, a warning is issued and CanNmPnInfoOffset is not set.</p> <p>CanNmPnInfo/CanNmPnInfoLength is set to PNC-VECTOR-LENGTH. If PNC-VECTOR-LENGTH is not defined, a warning is issued and CanNmPnInfoLength is not set.</p> <p>The values for CanNmPnFilterMaskByte are calculated as follows:</p> <p>In a first step the PNC-WAKEUP-DATA-MASK values of all CAN-COMMUNICATION-CONNECTOR elements that belong to the imported ECU-INSTANCE and a PHYSICAL-CHANNEL of the CAN-NM-CLUSTER's CAN-CLUSTER are retrieved. In the PNC vector which is a 64 bit entity, all bits that lie on any collected PNC-WAKEUP-DATA-MASK position are set to true, while all other bits are set to false. The resulting value is shifted to the right by PNC-VECTOR-OFFSET * 8 bit positions and thus removing the leading empty bits in the PNC vector. Finally, the value is converted into an n-byte byte array, (n corresponding to PNC-VECTOR-LENGTH).</p> <p>A CanNmPnFilterMaskByte container is created for each byte. CanNmPnFilterMaskByteValue is set to the value of the byte and CanNmPnFilterMaskByteIndex to the position of the byte within the array. In this array CanNmPnFilterMaskByteIndex of the PNC vector LSB is set to 0 and the CanNmPnFilterMaskByteIndex of the MSB is set to PNC-VECTOR-LENGTH - 1.</p>
CanNmChannelConfig	<p>For every CAN-NM-CLUSTER which belongs to the imported ECU-INSTANCE, and over which at least one NM-PDU is sent or received, a CanNmChannelConfig container is created.</p> <p>The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the CAN-NM-CLUSTER.</p> <p>A CAN-NM-CLUSTER belongs to the imported ECU-INSTANCE if at least one of its CAN-NM-NODE elements references a CAN-COMMUNICATION-CONTROLLER of this ECU-INSTANCE.</p> <p>The NM-ECU used for configuring some of the CanNmChannelConfig parameters is the NM-ECU which the first CAN-NM-NODE references via NM-IF-ECU-REF.</p>

Configuration parameters	Mapping description
	<p>In case the ECU-INSTANCE does not send or receive any NM-PDU on the CAN-NM-CLUSTER, or it sends multiple NM-PDU elements on the CAN-NM-CLUSTER, a warning is reported.</p> <p>CanNmNodeDetectionEnabled is set to NM-CLUSTER/NM-NODE-DETECTION-ENABLED, or to NM-ECU/NM-NODE-DETECTION-ENABLED if NM-CLUSTER/NM-NODE-DETECTION-ENABLED is not available.</p> <p>CanNmNodeIdEnabled is set to NM-CLUSTER/NM-NODE-ID-ENABLED, or to NM-ECU/NM-NODE-ID-ENABLED if NM-CLUSTER/NM-NODE-ID-ENABLED is not available.</p> <p>CanNmRepeatMsgIndEnabled is set to NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED, or to NM-ECU/NM-REPEAT-MSG-IND-ENABLED if NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED is not available.</p> <p>CanNmBusLoadReductionActive is set to NM-BUSLOAD-REDUCTION-ACTIVE.</p> <p>CanNmTimeoutTime is set to NM-NETWORK-TIMEOUT.</p> <p>CanNmWaitBusSleepTime is set to NM-WAIT-BUS-SLEEP-TIME.</p> <p>CanNmRepeatMessageTime is set to NM-REPEAT-MESSAGE-TIME.</p> <p>CanNmRemoteSleepIndTime is set to NM-REMOTE-SLEEP-INDICATION-TIME.</p> <p>CanNmUserDataLength is calculated by the formula <math>\langle \text{PayloadLength} \rangle - \langle \text{NonUserDataLength} \rangle</math> where <math>\langle \text{PayloadLength} \rangle</math> is calculated for the NM-PDU as described in <a href="#">Section 3.3.7, “PDU length calculation”</a> and <math>\langle \text{NonUserDataLength} \rangle</math> denotes the byte length of the non-user data part of the NM-PDU which is calculated by <math>\max(\text{CAN-NM-CLUSTER/NM-CBV-POSITION}, \text{CAN-NM-CLUSTER/NM-NID-POSITION}) + 1</math>. If neither CAN-NM-CLUSTER/NM-CBV-POSITION nor CAN-NM-CLUSTER/NM-NID-POSITION are defined, <math>\langle \text{NonUserDataLength} \rangle</math> is zero.</p> <p>CanNmMsgCycleTime is set to NM-MSG-CYCLE-TIME.</p> <p>CanNmMsgTimeoutTime is set to NM-MESSAGE-TIMEOUT-TIME.</p> <p>CanNmImmediateNmTransmissions is set to NM-IMMEDIATE-NM-TRANSMISSIONS.</p>

Configuration parameters	Mapping description
	<p><code>CanNmImmediateNmCycleTime</code> is set to <code>NM-IMMEDIATE-NM-CYCLE-TIME</code>.</p> <p><code>CanNmComMNetworkHandleRef</code> references the <code>ComMChannel</code> container that is created for the <code>COMMUNICATION-CLUSTER</code> referenced in <code>COMMUNICATION-CLUSTER-REF</code>.</p> <p><code>CanNmPduNidPosition</code> is set depending on the value of <code>NM-NID-POSITION</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>CANNM_PDU_BYTE_0</code> for 0.</li> <li>▶ <code>CANNM_PDU_BYTE_1</code> for 1.</li> <li>▶ <code>CANNM_PDU_OFF</code> for any other value.</li> </ul> <p><code>CanNmPduCbvPosition</code> is set depending on the value of <code>NM-CBV-POSITION</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>CANNM_PDU_BYTE_0</code> for 0.</li> <li>▶ <code>CANNM_PDU_BYTE_1</code> for 1.</li> <li>▶ <code>CANNM_PDU_OFF</code> for any other value.</li> </ul> <p><code>CanNmRxPdu/&lt;EcuC container name&gt;/CanNmRxPduRef</code> references the container in the <code>EcuC</code> module which has been created for the <code>NM-PDU</code> referenced via <code>RX-NM-PDU-REFS/RX-NM-PDU-REF</code>, which has the lowest <code>CAN-ID</code> assigned. <code>&lt;EcuC container name&gt;</code> is the name of the referenced <code>EcuC</code> container.</p> <p>If an <code>NmUserDataPdu</code> container has been created for the received <code>NM-PDU</code> in the <code>EcuC</code> module configuration (see <a href="#">Section 3.4.14, “EcuC”</a>), <code>CanNmUserDataRxPdu/CanNmRxUserDataPduRef</code> references this container. The name of <code>CanNmUserDataRxPdu</code> is set to the name of the referenced <code>EcuC</code> container.</p> <p><code>CanNmTxPdu/CanNmTxPduRef</code> references the container in the <code>EcuC</code> module which has been created for the first <code>NM-PDU</code> referenced via <code>TX-NM-PDU-REFS/TX-NM-PDU-REF</code>. The name of <code>CanNmTxPdu</code> is set to the name of the referenced <code>EcuC</code> container.</p> <p>If an <code>NmUserDataPdu</code> container has been created for the sent <code>NM-PDU</code> in the <code>EcuC</code> module configuration, see <a href="#">Section 3.4.14, “EcuC”</a>, <code>CanNmUserDataTxPdu/CanNmTxUserDataPduRef</code> references this container. The name of <code>CanNmUserDataTxPdu</code> is set to the name of the referenced <code>EcuC</code> container.</p> <p>If no Rx or Tx <code>NM-PDU</code> elements are referenced, or if multiple Tx <code>NM-PDU</code> elements are referenced, a warning is reported.</p>

Configuration parameters	Mapping description
	<p>If an ERA PDU container has been created for the CAN-NM-CLUSTER in the EcuC module (see <a href="#">Section 3.4.14, "EcuC"</a>), CanNmPnEraRxNSduRef references the EcuC container and CanNmPnEraCalcEnabled is set to true.</p> <p>The following parameters are set using the first NM-NODE of the CAN-NM-CLUSTER connected to the imported ECU-INSTANCE. If inconsistencies are detected among parameters of multiple NM-NODE elements, a warning is reported.</p> <p>CanNmMsgCycleOffset is set to NM-MSG-CYCLE-OFFSET.</p> <p>CanNmMsgReducedTime is set to NM-MSG-REDUCED-TIME.</p> <p>CanNmNodeId is set to NM-NODE-ID.</p> <p>CanNmPnEnabled is set to true if the CAN-NM-CLUSTER is PNC-enabled, or if it has its NM-PNC-PARTICIPATION either not defined or set to true. Otherwise CanNmPnEnabled is set to false.</p> <p>If the CAN-NM-CLUSTER is PNC-enabled, the following parameters are set:</p> <p>CanNmCarWakeUpRxEnabled is set to NM-CAR-WAKE-UP-RX-ENABLED.</p> <p>CanNmCarWakeUpFilterEnabled is set to NM-CAR-WAKE-UP-FILTER-ENABLED.</p> <p>CanNmCarWakeUpFilterNodeId is set to NM-CAR-WAKE-UP-FILTER-NODE-ID.</p> <p>CanNmCarWakeUpBitPosition is set to NM-CAR-WAKE-UP-BIT-POSITION mod 8.</p> <p>CanNmCarWakeUpBytePosition is set to NM-CAR-WAKE-UP-BIT-POSITION div 8.</p>

### 3.4.5. CanSM

Configuration parameters	Mapping description
CanSMConfiguration/CanSMManagerNetwork	For every CAN-CLUSTER which belongs to the imported ECU-INSTANCE, a CanSMManagerNetwork container is created. The container name is <PREFIX><name>, where <name> is the SHORT-NAME of the CAN-CLUSTER.

Configuration parameters	Mapping description
	CanSMComMNetworkHandleRef references the ComMChannel container in the ComM module configuration, which has been created for the CAN-CLUSTER.
CanSMConfiguration/CanSMManager-Network/CanSMController	<p>For every CAN-COMMUNICATION-CONTROLLER that is connected to the CAN-CLUSTER and to the imported ECU-INSTANCE, a CanSMController container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the CAN-COMMUNICATION-CONTROLLER.</p> <p>CanSMControllerId references the CanIfCtrlCfg container in the CanIf module configuration, which has been created for the CAN-COMMUNICATION-CONTROLLER.</p>

### 3.4.6. CanTp

Configuration parameters	Mapping description
CanTpConfig	CanTpMainFunctionPeriod is set to CYCLE-TIME-MAIN-FUNCTION of the first CAN-TP-ECU which belongs to the imported ECU-INSTANCE. If inconsistencies are detected among the CYCLE-TIME-MAIN-FUNCTION values of multiple CAN-TP-ECU elements of the imported ECU-INSTANCE, a warning is reported.
CanTpConfig/CanTpChannel	<p>For each CAN-TP-CHANNEL referenced by a CAN-TP-CONNECTION which belongs to the imported ECU-INSTANCE, a CanTpChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the CAN-TP-CHANNEL. A CAN-TP-CONNECTION belongs to the imported ECU-INSTANCE if at least one transmitting or receiving CAN-TP-NODE of this CAN-TP-CONNECTION references a COMMUNICATION-CONNECTOR which is also referenced by the imported ECU-INSTANCE.</p> <p>CanTpChannelMode is set depending on the value of CHANNEL-MODE:</p> <ul style="list-style-type: none"> <li>► CANTP_MODE_FULL_DUPLEX for FULLDUPLEXMODE.</li> <li>► CANTP_MODE_HALF_DUPLEX for HALFDUPLEXMODE.</li> </ul>
CanTpConfig/CanTpChannel/CanTpRxNSdu	For every CAN-TP-CONNECTION referencing the current CAN-TP-CHANNEL, a CanTpRxNSdu container is created, if the PDU referenced via TP-SDU-REF is received or routed (see <a href="#">Section 3.3.6, "PDU routing"</a> ) by the imported ECU-INSTANCE. The container name is <PREFIX><name>, where <name> is the SHORT-NAME of the referenced PDU.



Configuration parameters	Mapping description
	<p>CanTpRxNSduRef references the container in the EcuC module configuration, created for the referenced PDU.</p> <p>CanTpNTa/CanTpNTa is set to TP-ADDRESS of the CAN-TP-ADDRESS referenced via MULTICAST-REF. If no MULTICAST-REF exists, the CAN-TP-ADDRESS of the first CAN-TP-NODE (if more than one exists, a warning is issued) referenced via RECEIVER-REFS is used. Furthermore, CanTpNTa is only set if ADDRESSING-FORMAT of the CAN-TP-CONNECTION is set to EXTENDED.</p> <p>CanTpNSa/CanTpNSa is set to TP-ADDRESS of the CAN-TP-ADDRESS referenced by the CAN-TP-NODE referenced via TRANSMITTER-REF. CanTpNSa is only set if ADDRESSING-FORMAT of the CAN-TP-CONNECTION is set to EXTENDED.</p> <p>CanTpNAe/CanTpNAe is set to TP-ADDRESS-EXTENSION-VALUE of the CAN-TP-ADDRESS referenced via MULTICAST-REF. If no MULTICAST-REF exists, the CAN-TP-ADDRESS of the first CAN-TP-NODE referenced via RECEIVER-REFS is used. Additionally the same TP-ADDRESS-EXTENSION-VALUE must exist for the CAN-TP-NODE referenced via TRANSMITTER-REF. Otherwise a warning is issued and the parameter is not set. This parameter is only configured if ADDRESSING-FORMAT of the CAN-TP-CONNECTION is set to MIXED.</p> <p>CanTpRxPaddingActivation is set to CANTP_ON if PADDING-ACTIVATION is set to true, or to CANTP_OFF if PADDING-ACTIVATION is set to false.</p> <p>CanTpRxTaType is set depending on the value of TA-TYPE:</p> <ul style="list-style-type: none"> <li>▶ CANTP_FUNCTIONAL for CANTP-FUNCTIONAL.</li> <li>▶ CANTP_PHYSICAL for CANTP-PHYSICAL.</li> </ul> <p>If TA-TYPE is not available CanTpRxTaType is set to CANTP_FUNCTIONAL if a CAN-TP-ADDRESS referenced via MULTICAST-REF exists, or if more than one CAN-TP-CONNECTION elements references the same PDU via TP-SDU-REF. In all other cases, it is set to CANTP_PHYSICAL.</p> <p>CanTpRxAddressingFormat is set depending on the value of ADDRESSING-FORMAT:</p> <ul style="list-style-type: none"> <li>▶ CANTP_STANDARD for STANDARD.</li> <li>▶ CANTP_EXTENDED for EXTENDED.</li> <li>▶ CANTP_MIXED for MIXED.</li> </ul>

Configuration parameters	Mapping description
	<p>CanTpBs is set to MAX-BLOCK-SIZE.</p> <p>CanTpNbr is set to TIMEOUT-BR.</p> <p>CanTpNcr is set to TIMEOUT-CR.</p> <p>If no MULTICAST-REF exists, and if exactly one CAN-TP-NODE is referenced via RECEIVER-REFS, the following parameters are set using this CAN-TP-NODE.</p> <p>CanTpRxWftMax is set to MAX-FC-WAIT.</p> <p>CanTpNar is set to TIMEOUT-AR.</p> <p>CanTpSTmin is set to ST-MIN .</p>
CanTpConfig/CanTpChannel/CanTpRxNSdu/CanTpRxNPdu	<p>A CanTpRxNPdu container is created for the N-PDU referenced via DATA-PDU-REF. If no DATA-PDU-REF exists, a warning is issued. The name of CanTpRxNPdu is set to &lt;SDU name&gt;_ &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;SDU name&gt; is the name of the parent CanTpRxNSdu container and &lt;name&gt; is the SHORT-NAME of the N-PDU.</p> <p>CanTpRxNPduRef references the corresponding container in the EcuC module configuration.</p>
CanTpConfig/CanTpChannel/CanTpTxNSdu/CanTpTxFCNPdu	<p>A CanTpTxFCNPdu container is created for the N-PDU referenced via FLOW-CONTROL-PDU-REF. The name of CanTpTxFCNPdu is set to &lt;SDU name&gt;_ &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;SDU name&gt; is the name of the parent CanTpTxNSdu container and &lt;name&gt; is the SHORT-NAME of the N-PDU.</p> <p>CanTpTxFCNPduRef references the corresponding container in the EcuC module configuration.</p>
CanTpConfig/CanTpChannel/CanTpTxNSdu	<p>For every CAN-TP-CONNECTION referencing the current CAN-TP-CHANNEL, a CanTpTxNSdu container is created, if the PDU referenced via TP-SDU-REF is sent or routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) by the imported ECU-INSTANCE. The container name is &lt;SDU name&gt;_ &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;SDU name&gt; is the name of the parent CanTpTxNSdu container and &lt;name&gt; is the SHORT-NAME of the referenced PDU.</p> <p>In case multiple CAN-TP-CONNECTION elements refer to the same PDU via TP-SDU-REF, only one CanTpTxNSdu container is created and the following parameters of the CAN-TP-CONNECTION are checked for consistency: DATA-PDU-REF, FLOW-CONTROL-PDU-REF, ADDRESSING-FORMAT, PADDING-ACTI-</p>

Configuration parameters	Mapping description
	<p>VATION, TRANSMITTER-REF. If inconsistencies are detected a warning is issued and the first CAN-TP-CONNECTION is used for the subsequent operations.</p> <p>CanTpTxNSduRef: <b>see</b> CanTpRxNSdu/CanTpRxNSduRef.</p> <p>CanTpNTa: <b>see</b> CanTpRxNSdu/CanTpNTa. If multiple CAN-TP-CONNECTION elements refer to the same PDU via TP-SDU-REF, and the TP-ADDRESS values of the first CAN-TP-NODE of each CAN-TP-CONNECTION referenced via RECEIVER-REF differ, CanTpNTa is not set.</p> <p>CanTpNSa: <b>see</b> CanTpRxNSdu/CanTpNSa.</p> <p>CanTpNAe: <b>see</b> CanTpRxNSdu/CanTpNAe.</p> <p>CanTpTxPaddingActivation: <b>see</b> CanTpRxNSdu/CanTpRxPaddingActivation.</p> <p>CanTpTxTaType: <b>see</b> CanTpRxNSdu/CanTpRxTaType.</p> <p>CanTpNbs is set to TIMEOUT-BS.</p> <p>CanTpNcs is set to TIMEOUT-CS.</p> <p>CanTpTc is set to CANCELLATION.</p> <p>CanTpNas is set to TIMEOUT-AS of the CAN-TP-NODE referenced via TRANSMITTER-REF.</p>
CanTpConfig/CanTpChannel/CanTpTxNSdu/CanTpTxNPdu	<p><b>See</b> CanTpRxNSdu/CanTpRxNPdu.</p> <p>In case a received N-PDU is referenced by two CAN-TP-CONNECTION elements of the imported ECU-INSTANCE (either via DATA-PDU-REF by the CAN-TP-CONNECTION referencing a Tx PDU via TP-SDU-REF, or via FLOW-CONTROL-PDU-REF by the CAN-TP-CONNECTION referencing an Rx PDU via TP-SDU-REF), CanTpTxNPduRef references the duplicate container in the EcuC module configuration (container name: &lt;PREFIX&gt;&lt;name&gt;_D&lt;INSTSUFFIX&gt;).</p>
CanTpConfig/CanTpChannel/CanTpTxNSdu/CanTpRxFcNPdu	<p><b>See</b> CanTpRxNSdu/CanTpTxFcNPdu.</p>

### 3.4.7. CanTSyn

Configuration parameters	Mapping description
CanTSynGlobalTimeDomain	<p>One CanTSynGlobalTimeDomain container is created for each GLOBAL-TIME-DOMAIN that contains a GLOBAL-TIME-CAN-MASTER or a GLOBAL-TIME-CAN-SLAVE element that references a CAN-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-DOMAIN.</p> <p>CanTSynGlobalTimeDomainId is set to DOMAIN-ID.</p> <p>CanTSynSynchronizedTimeBaseRef references the corresponding StbMSynchronizedTimeBase container in the StbM module. If the GLOBAL-TIME-DOMAIN is a subdomain of a parent GLOBAL-TIME-DOMAIN, CanTSynSynchronizedTimeBaseRef references the StbMSynchronizedTimeBase container that has been created for the parent GLOBAL-TIME-DOMAIN. For further information about the configuration of the StbM module, see <a href="#">Section 3.4.45, “StbM”</a>.</p>
CanTSynGlobalTimeDomain/CanTSynGlobalTimeFupDataIDList, CanTSynGlobalTimeDomain/CanTSynGlobalTimeOfnsDataIDList, CanTSynGlobalTimeDomain/CanTSynGlobalTimeOfsDataIDList, CanTSynGlobalTimeDomain/CanTSynGlobalTimeSyncDataIDList	<p>The CAN-GLOBAL-TIME-DOMAIN-PROPS element of the CAN GLOBAL-TIME-DOMAIN entity is used to retrieve the following subelements to configure container lists:</p> <ul style="list-style-type: none"> <li>▶ FUP-DATA-ID-LIST</li> <li>▶ OFNS-DATA-ID-LIST</li> <li>▶ OFS-DATA-ID-LIST</li> <li>▶ SYNC-DATA-ID-LIST</li> </ul> <p>The configured container lists are the following:</p> <ul style="list-style-type: none"> <li>▶ CanTSynGlobalTimeDomain/CanTSynGlobalTimeFupDataIDList/CanTSynGlobalTimeFupDataIDListElement</li> <li>▶ CanTSynGlobalTimeDomain/CanTSynGlobalTimeOfnsDataIDList/CanTSynGlobalTimeOfnsDataIDListElement</li> <li>▶ CanTSynGlobalTimeDomain/CanTSynGlobalTimeOfsDataIDList/CanTSynGlobalTimeOfsDataIDListElement</li> <li>▶ CanTSynGlobalTimeDomain/CanTSynGlobalTimeSyncDataIDList/CanTSynGlobalTimeSyncDataIDListElement</li> </ul>

Configuration parameters	Mapping description															
	<p>In each container list, one subcontainer is created per element named <code>Element_&lt;idx&gt;</code>, where <code>&lt;idx&gt;</code> is the zero-based index of the element within the list. In each of these lists, one index parameter and one value parameter are configured. The index parameter represents the zero-based index of the element in the list, the value parameter represents the actual value.</p> <table><tr><th>ID List</th><th>Index Parameter</th><th>Value Parameter</th></tr><tr><td>CanTSynGlobalTimeFupDataIDList</td><td>CanTSynGlobalTimeFupDataIDListIndex</td><td>CanTSynGlobalTimeFupDataIDListValue</td></tr><tr><td>CanTSynGlobalTimeOfnsDataIDList</td><td>CanTSynGlobalTimeOfnsDataIDListIndex</td><td>CanTSynGlobalTimeOfnsDataIDListValue</td></tr><tr><td>CanTSynGlobalTimeOfsDataIDList</td><td>CanTSynGlobalTimeOfsDataIDListIndex</td><td>CanTSynGlobalTimeOfsDataIDListValue</td></tr><tr><td>CanTSynGlobalTimeSyncDataIDList</td><td>CanTSynGlobalTimeSyncDataIDListIndex</td><td>CanTSynGlobalTimeSyncDataIDListValue</td></tr></table>	ID List	Index Parameter	Value Parameter	CanTSynGlobalTimeFupDataIDList	CanTSynGlobalTimeFupDataIDListIndex	CanTSynGlobalTimeFupDataIDListValue	CanTSynGlobalTimeOfnsDataIDList	CanTSynGlobalTimeOfnsDataIDListIndex	CanTSynGlobalTimeOfnsDataIDListValue	CanTSynGlobalTimeOfsDataIDList	CanTSynGlobalTimeOfsDataIDListIndex	CanTSynGlobalTimeOfsDataIDListValue	CanTSynGlobalTimeSyncDataIDList	CanTSynGlobalTimeSyncDataIDListIndex	CanTSynGlobalTimeSyncDataIDListValue
ID List	Index Parameter	Value Parameter														
CanTSynGlobalTimeFupDataIDList	CanTSynGlobalTimeFupDataIDListIndex	CanTSynGlobalTimeFupDataIDListValue														
CanTSynGlobalTimeOfnsDataIDList	CanTSynGlobalTimeOfnsDataIDListIndex	CanTSynGlobalTimeOfnsDataIDListValue														
CanTSynGlobalTimeOfsDataIDList	CanTSynGlobalTimeOfsDataIDListIndex	CanTSynGlobalTimeOfsDataIDListValue														
CanTSynGlobalTimeSyncDataIDList	CanTSynGlobalTimeSyncDataIDListIndex	CanTSynGlobalTimeSyncDataIDListValue														
CanTSynGlobalTimeDomain/CanTSynGlobalTimeMaster	<p><b>If the GLOBAL-TIME-CAN-MASTER of the GLOBAL-TIME-DOMAIN references a CAN-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE, a CanTSynGlobalTimeMaster container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-CAN-MASTER.</b></p> <p>CanTSynGlobalTimeTx_crcSecured is set depending on the value of GLOBAL-TIME-CAN-MASTER/CRC-SECURED: CRC_SUPPORTED for CRC-SUPPORTED and CRC_NOT_SUPPORTED for CRC-NOT-SUPPORTED.</p> <p>CanTSynGlobalTimeTxFollowUpOffset is set to GLOBAL-TIME-CAN-MASTER/FOLLOW-UP-OFFSET.</p> <p>CanTSynGlobalTimeTxPeriod is set to GLOBAL-TIME-CAN-MASTER/SYNC-PERIOD.</p> <p>CanTSynMasterConfirmationTimeout is set to GLOBAL-TIME-CAN-MASTER/SYNC-CONFIRMATION-TIMEOUT.</p> <p>CanTSynCyclicMsgResumeTime is set to GLOBAL-TIME-CAN-MASTER/IMMEDIATE-RESUME-TIME.</p> <p>CanTSynGlobalTimeDebounceTime is set to GLOBAL-TIME-DOMAIN/DEBOUNCE-TIME.</p>															

Configuration parameters	Mapping description
CanTSynGlobalTime-Domain/CanTSynGlobalTimeMaster/CanTSynGlobalTimeMasterPdu	<p>A GLOBAL-TIME-DOMAIN is associated with a PDU if that PDU is a GENERAL-PURPOSE-PDU that has its CATEGORY set to GLOBAL_TIME and one of the conditions hold:</p> <ul style="list-style-type: none"> <li>▶ The GLOBAL-TIME-DOMAIN references the PDU in GLOBAL-TIME-PDU-REF</li> <li>▶ The GLOBAL-TIME-DOMAIN references a PDU-TRIGGERING either via GLOBAL-TIME-PDU-TRIGGERING-REF or via PDU-TRIGGERING-REF, and that PDU-TRIGGERING refers to the PDU</li> </ul> <p>If the GLOBAL-TIME-DOMAIN is associated with a PDU, and if this is the PDU that the configured ECU-INSTANCE sends on the CAN-CLUSTER which the GLOBAL-TIME-DOMAIN references via COMMUNICATION-CLUSTER-REF, one CanTSynGlobalTimeMasterPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>CanTSynGlobalTimePduRef references the EcuC container created for the PDU that is associated with the GLOBAL-TIME-DOMAIN.</p>
CanTSynGlobalTime-Domain/CanTSynGlobalTimeSlave	<p>If one of the GLOBAL-TIME-CAN-SLAVE elements of the GLOBAL-TIME-DOMAIN references a CAN-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE, a CanTSynGlobalTimeSlave container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-CAN-SLAVE.</p> <p>CanTSynRxCrcValidated is set depending on the value of GLOBAL-TIME-CAN-SLAVE/CRC-VALIDATED: CRC_VALIDATED for CRC-VALIDATED, CRC_NOT_VALIDATED for CRC-NOT-VALIDATED, and CRC_IGNORED for CRC-IGNORED.</p> <p>CanTSynGlobalTimeFollowUpTimeout is set to GLOBAL-TIME-CAN-SLAVE/FOLLOW-UP-TIMEOUT-VALUE or to GLOBAL-TIME-DOMAIN/FOLLOW-UP-TIMEOUT-VALUE if GLOBAL-TIME-CAN-SLAVE/FOLLOW-UP-TIMEOUT-VALUE does not exist.</p> <p>CanTSynGlobalTimeSequenceCounterJumpWidth is set to GLOBAL-TIME-CAN-SLAVE/SEQUENCE-COUNTER-JUMP-WIDTH.</p>
CanTSynGlobalTime-Domain/CanTSynGlobalTimeSlave/Can-	<p>If the GLOBAL-TIME-DOMAIN is associated with a PDU, and if this is the PDU that the configured ECU-INSTANCE and if this is the PDU that the configured ECU-INSTANCE receives on the CAN-CLUSTER which the GLOB-</p>

Configuration parameters	Mapping description
TSynGlobal-TimeSlavePdu	<p>AL-TIME-DOMAIN references via COMMUNICATION-CLUSTER-REF, one Can-TSynGlobalTimeSlavePdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>CanTSynGlobalTimePduRef references the EcuC container created for the PDU that is associated with the GLOBAL-TIME-DOMAIN.</p>

### 3.4.8. Com

Configuration parameters	Mapping description
ComGeneral	ComEnableMDTForCyclicTransmission is set to COM-ENABLE-MDT-FOR-CYCLIC-TRANSMISSION.
ComConfig/ComIPdu	<p>For every PDU instance (see <a href="#">Section 3.3.1.3, "Instance handling"</a>) which represents an I-SIGNAL-I-PDU or an NM-PDU for which an NmUserDataPdu container has been created (see <a href="#">Section 3.4.14, "EcuC"</a>), a ComIPdu container is created, except if an LdCom module configuration is present in the current project and the conditions described in <a href="#">Section 3.4.34, "LdCom"</a> allow to process the PDU instance in the LdCom module. For I-SIGNAL-I-PDU elements the container name is PD&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-I-PDU. For NM-PDU elements the container name is PD&lt;PREFIX&gt;&lt;name&gt;_NmComUserData&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the NM-PDU.</p> <p>The ComIPduSignalRef entries reference all ComSignal containers representing signal instances of the PDU instance which lie directly within the PDU (in contrast to group signals embedded in PDUs indirectly via signal groups). See also <a href="#">Section 3.3.1.3, "Instance handling"</a>.</p> <p>The ComIPduSignalGroupRef entries reference all ComSignalGroup containers which have been created for the signal group instances of the PDU instance.</p> <p>ComIPduDirection is set to SEND if the PDU instance is sent by the imported ECU-INSTANCE. Otherwise it is set to RECEIVE.</p> <p>ComIPduType is set to TP if one of the following conditions holds:</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ The I-SIGNAL-I-PDU is referenced by a CAN-TP-CONNECTION via TP-SDU-REF</li> <li>▶ The I-SIGNAL-I-PDU is referenced by a FLEXRAY-TP-CONNECTION via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF</li> <li>▶ The I-SIGNAL-I-PDU is referenced by a FLEXRAY-AR-TP-CONNECTION via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF</li> <li>▶ The I-SIGNAL-I-PDU is referenced by a LIN-TP-CONNECTION via LIN-TP-N-SDU-REF</li> <li>▶ The I-SIGNAL-I-PDU is referenced by a PDU-TRIGGERING, which in turn is referenced by a SOMEIP-TP-CONNECTION via TP-SDU-REF</li> <li>▶ An ETH-TP-CONNECTION refers to the PDU-TRIGGERING that references the transmitted I-SIGNAL-I-PDU and that belongs to the PHYSICAL-CHANNEL on which the I-SIGNAL-I-PDU is transmitted.</li> <li>▶ One of the TP-CONNECTION elements listed above references a SECURED-I-PDU, either directly or via a PDU-TRIGGERING, and the SECURED-I-PDU in turn references the PDU-TRIGGERING of the I-SIGNAL-I-PDU via PAYLOAD-REF.</li> </ul> <p>In all other cases ComIPduType is set to NORMAL.</p> <p>The ComIPduGroupRef entries reference all ComIPduGroup containers which have been created for the I-SIGNAL-I-PDU-GROUP elements that contain the PDU instance's I-SIGNAL-I-PDU or NM-PDU. A ComIPduGroupRef entry is added as long as the COMMUNICATION-DIRECTION of the I-SIGNAL-I-PDU-GROUP is not opposite to the ComIPduDirection of the referencing I-SIGNAL-I-PDU. In case the I-SIGNAL-I-PDU is not part of any I-SIGNAL-I-PDU-GROUP, a reference to the default ComIPduGroup (see ComConfig/ComIPduGroup) this PDU belongs to is added. NM-PDU elements always obtain a reference to the default Nm ComIPduGroup, since I-SIGNAL-I-PDU-GROUP elements cannot contain NM-PDU elements.</p> <p>ComIPduSignalProcessing is set to I-PDU-SIGNAL-PROCESSING of the I-PDU-PORT that connects one of the COMMUNICATION-CONNECTOR elements of the ECU-INSTANCE to the PDU-TRIGGERING of the PDU instance.</p> <p>ComPduIdRef references the container created for the I-SIGNAL-I-PDU in the EcuC module configuration. For NM-PDU elements it references the NmUserDataPdu container.</p>



Configuration parameters	Mapping description
ComConfig/ComIPdu/ComTxIPdu	<p>If the PDU instance is sent by the imported ECU-INSTANCE, a ComTxIPdu sub container is created.</p> <p>ComTxIPduUnusedAreasDefault is set to UNUSED-BIT-PATTERN. If UNUSED-BIT-PATTERN does not lie within the interval [0 .. 255] a warning is reported and ComTxIPduUnusedAreasDefault is not configured.</p> <p>ComMinimumDelayTime is set to MINIMUM-DELAY of the I-SIGNAL-I-PDU's I-PDU-TIMING.</p>
ComConfig/ComIPdu/ComTxIPdu/ComTxModeTrue/ComTxMode	<p>For every sent PDU instance, a ComTxModeTrue/ComTxMode container is created. For NM-PDU elements and for I-SIGNAL-I-PDU elements without TRANSMISSION-MODE-TRUE-TIMING, ComTxModeMode is set to NONE.</p> <p>If TRANSMISSION-MODE-TRUE-TIMING exists for the PDU instance, ComTxModeMode is set depending on the existence of timings attached to the TRANSMISSION-MODE-TRUE-TIMING:</p> <ul style="list-style-type: none"> <li>▶ MIXED, if both CYCLIC-TIMING and EVENT-CONTROLLED-TIMING exist.</li> <li>▶ PERIODIC, if only a CYCLIC-TIMING exists.</li> <li>▶ DIRECT, if only an EVENT-CONTROLLED-TIMING exists.</li> <li>▶ NONE, if no timing exists.</li> </ul> <p>For NM-PDU elements ComTxModeMode is always set to NONE.</p> <p>ComTxModeTimePeriod is set to TIME-PERIOD/VALUE of the CYCLIC-TIMING if ComTxModeMode is set to MIXED or PERIODIC.</p> <p>ComTxModeTimeOffset is set to TIME-OFFSET/VALUE of the CYCLIC-TIMING if ComTxModeMode is set to MIXED or PERIODIC.</p> <p>ComTxModeNumberOfRepetitions is set to (NUMBER-OF-REPETITIONS + 1) of the EVENT-CONTROLLED-TIMING if ComTxModeMode is set to MIXED or DIRECT. If NUMBER-OF-REPETITIONS equals 0, ComTxModeNumberOfRepetitions is also set to 0.</p> <p>ComTxModeRepetitionPeriod is set to REPETITION-PERIOD/VALUE of the EVENT-CONTROLLED-TIMING if ComTxModeMode is set to MIXED or DIRECT and if NUMBER-OF-REPETITIONS of the EVENT-CONTROLLED-TIMING is greater than zero.</p>

Configuration parameters	Mapping description
ComConfig/ComIPdu/ComTxIPdu/ComTxModeFalse/ComTxMode	<p>If a TRANSMISSION-MODE-FALSE-TIMING exists for the I-SIGNAL-I-PDU, a ComTxModeFalse/ComTxMode container is created.</p> <p>For the configuration of the container parameters see ComTxMode-True/ComTxMode.</p>
ComConfig/ComSignal	<p>For every signal instance which represents an I-SIGNAL which is sent or received by the imported ECU-INSTANCE and which is not contained in any I-SIGNAL-GROUP, a ComISignal container is created. A signal instance is considered sent if there is an I-SIGNAL-PORT with COMMUNICATION-DIRECTION set to OUT that connects one of the COMMUNICATION-CONNECTOR elements of the imported ECU-INSTANCE to the I-SIGNAL-TRIGGERING of the signal instance. If COMMUNICATION-DIRECTION is set to IN, the signal instance is considered received. The container name is SG&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL. In special cases, dedicated IRA signal containers are added. The configuration of IRA signals is described in <a href="#">Section 3.4.8.2, "Configuration for partial networking IRA signals"</a>.</p> <p>ComTimeout is set to TIMEOUT of the I-SIGNAL-PORT that connects one of the COMMUNICATION-CONNECTOR elements of the ECU-INSTANCE to the I-SIGNAL-TRIGGERING of the signal instance.</p> <p>ComFirstTimeout is set to FIRST-TIMEOUT of the I-SIGNAL-PORT that connects one of the COMMUNICATION-CONNECTOR elements of the ECU-INSTANCE to the I-SIGNAL-TRIGGERING of the signal instance.</p> <p>ComDataInvalidAction is set according to the value of HANDLE-INVALID of the I-SIGNAL-PORT that connects one of the COMMUNICATION-CONNECTOR elements of the ECU-INSTANCE to the I-SIGNAL-TRIGGERING of the signal instance:</p> <ul style="list-style-type: none"> <li>▶ NOTIFY for KEEP.</li> <li>▶ REPLACE for REPLACE.</li> </ul> <p>Other values of HANDLE-INVALID are ignored.</p> <p>ComUpdateBitPosition is set to UPDATE-INDICATION-BIT-POSITION of the signal instance's I-SIGNAL-TO-I-PDU-MAPPING.</p> <p>ComTransferProperty is set depending on the value of I-SIGNAL-TO-I-PDU-MAPPING/TRANSFER-PROPERTY:</p> <ul style="list-style-type: none"> <li>▶ PENDING for PENDING.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ TRIGGERED <b>for</b> TRIGGERED.</li> <li>▶ TRIGGERED_ON_CHANGE <b>for</b> TRIGGERED-ON-CHANGE.</li> <li>▶ TRIGGERED_ON_CHANGE_WITHOUT_REPETITION <b>for</b> TRIGGERED-ON-CHANGE-WITHOUT-REPETITION.</li> <li>▶ TRIGGERED_WITHOUT_REPETITION <b>for</b> TRIGGERED-WITHOUT-REPETITION.</li> </ul> <p>For I-SIGNAL elements of NM-PDU elements, ComTransferProperty is always set to PENDING.</p> <p>ComSystemTemplateSystemSignalRef is set to the AUTOSAR path of the I-SIGNAL-TO-I-PDU-MAPPING of the signal instance.</p> <p>ComSignalInitValue is set to VALUE of the INIT-VALUE of the I-SIGNAL.</p> <p>Depending on the type of VALUE-SPECIFICATION the INIT-VALUE must be converted first:</p> <ul style="list-style-type: none"> <li>▶ NUMERICAL-VALUE-SPECIFICATION <p>If the ComSignalType is UINT8_N or UINT8_DYN, the value provided in NUMERICAL-VALUE-SPECIFICATION is converted to a string which represents a byte array separated by spaces. If the value does not represent an integral value, a warning is issued and ComSignalInitValue is not set. Otherwise the value provided in NUMERICAL-VALUE-SPECIFICATION is configured without conversion.</p> </li> <li>▶ ARRAY-VALUE-SPECIFICATION <p>If the ComSignalType is UINT8_N or UINT8_DYN, the ARRAY-VALUE-SPECIFICATION is converted to a string which represents a byte array separated by spaces.</p> <p>Otherwise, the ARRAY-VALUE-SPECIFICATION is converted to an integral number where the number's most significant byte contains the first element of the ARRAY-VALUE-SPECIFICATION.</p> </li> <li>▶ TEXT-VALUE-SPECIFICATION <p>If the ComSignalType is UINT8_N or UINT8_DYN, the TEXT-VALUE-SPECIFICATION is converted to a string which represents a byte array separated by spaces. Characters &gt; 255 are not supported and generate a warning.</p> </li> </ul>

Configuration parameters	Mapping description
	<p>Otherwise, the characters of the string are converted to an integral number where the number's most significant byte contains the first byte of the TEXT-VALUE-SPECIFICATION. Characters &gt; 127 generate a warning. If the string has a length of more than eight characters, an error is issued.</p> <p>► CONSTANT-REFERENCE</p> <p>A CONSTANT-REFERENCE refers to a CONSTANT-SPECIFICATION which in turn contains a VALUE-SPECIFICATION. This VALUE-SPECIFICATION is taken as input for configuring ComSignalInitValue.</p> <p>Other VALUE-SPECIFICATION elements are not supported and result in a warning and ComSignalInitValue not being set.</p> <p>ComSignalDataInvalidValue is set to VALUE of the INVALID-VALUE of the I-SIGNAL's NETWORK-REPRESENTATION-PROPS.</p> <p>Depending on the type of VALUE-SPECIFICATION the INVALID-VALUE is converted first, see ComSignalInitValue for conversion rules.</p> <p>ComBitPosition is set to START-POSITION of the signal instance's I-SIGNAL-TO-I-PDU-MAPPING.</p> <p>In case the ComSignalType determined for the I-SIGNAL is UINT8_N or UINT8_DYN, and the PACKING-BYTE-ORDER of the I-SIGNAL-TO-I-PDU-MAPPING is MOST-SIGNIFICANT-BYTE-FIRST, the START-POSITION needs to be converted to MOST-SIGNIFICANT-BYTE-LAST ("little endian"). If the converted position is not byte-aligned within the parent PDU, a warning is reported and the original value of START-POSITION is used.</p> <p>For I-SIGNAL elements of NM-PDU elements the resulting value of START-POSITION needs to be adjusted by the length of the non-user data (see PduLength in <a href="#">Section 3.4.14, "EcuC"</a>): <math>\text{ComBitPosition} = (\text{converted}) \text{ START-POSITION} - \langle \text{length of non-user data [bytes]} \rangle * 8</math>. If the value of ComBitPosition is less than zero, a warning is reported.</p> <p>ComSignalEndianness is set depending on the value of PACKING-BYTE-ORDER of the signal instance's I-SIGNAL-TO-I-PDU-MAPPING:</p> <p>► BIG_ENDIAN for MOST-SIGNIFICANT-BYTE-FIRST.</p> <p>► LITTLE_ENDIAN for MOST-SIGNIFICANT-BYTE-LAST.</p>

Configuration parameters	Mapping description
	<p>In case the <code>ComSignalType</code> determined for the I-SIGNAL is <code>UINT8_N</code> or <code>UINT8_DYN</code>, <code>ComSignalEndianness</code> is set to <code>OPAQUE</code>.</p> <p>The configuration of <code>ComBitSize</code> and <code>ComSignalLength</code> depends on what has been configured for <code>ComSignalType</code>:</p> <ul style="list-style-type: none"> <li>▶ If <code>ComSignalType</code> was set to <code>UINT8_N</code>, the value is taken from <code>BASE-TYPE-SIZE</code> of <code>SW-BASE-TYPE</code>. If <code>BASE-TYPE-SIZE</code> is not available, the <code>LENGTH</code> value of I-SIGNAL is taken. If <code>LENGTH</code> is not available the SDG named "<code>cme:ISignalExtension/cme:length</code>" is used instead.</li> <li>▶ In any other cases, the <code>LENGTH</code> value of I-SIGNAL is taken, or, if that value is not available, the SDG named "<code>cme:ISignalExtension/cme:length</code>" is used.</li> </ul> <p><code>ComBitSize</code> is set to the value calculated as described above. For <code>UINT8_N</code> or <code>UINT8_DYN</code> the parameter is only configured if it is a multiple of eight.</p> <p><code>ComSignalDirection</code> is set to <code>SEND</code> if the PDU instance is sent by the imported ECU-INSTANCE. Otherwise it is set to <code>RECEIVE</code>.</p> <p><code>ComSignalLength</code> is configured for <code>UINT8_N</code> or <code>UINT8_DYN</code> signals. If the value calculated as described above is divisible by eight without remainder, <code>ComSignalLength</code> is configured with the result of the division.</p>
<p><code>ComConfig/ComSignal/ComFilter</code></p>	<p>If a <code>TRANSMISSION-MODE-CONDITION</code> (I-SIGNAL-I-PDU -&gt; I-PDU-TIMING -&gt; <code>TRANSMISSION-MODE-DECLARATION</code> -&gt; <code>TRANSMISSION-MODE-CONDITION</code>) referring to the same I-SIGNAL-TO-I-PDU-MAPPING as the signal instance exists, and the signal instance is sent by the imported ECU-INSTANCE, a <code>ComFilter</code> container is created.</p> <p>If the <code>DATA-FILTER-TYPE</code> of the <code>TRANSMISSION-MODE-CONDITION</code>'s <code>DATA-FILTER</code> is set to <code>ALWAYS</code> and a <code>TRANSMISSION-MODE-TRUE-TIMING</code> exists for this <code>TRANSMISSION-MODE-CONDITION</code>, no <code>ComFilter</code> container is created. If in this case a <code>TRANSMISSION-MODE-FALSE-TIMING</code> exists for the <code>TRANSMISSION-MODE-CONDITION</code> as well, a warning is issued.</p> <p>If the I-SIGNAL-TO-I-PDU-MAPPING has already been referenced by another <code>TRANSMISSION-MODE-CONDITION</code>, a warning is issued and the <code>ComFilter</code> parameters are not exported a second time.</p> <p>A <code>ComFilter</code> container is also created for received signal instances, if a <code>DATA-FILTER</code> exists for the I-SIGNAL-PORT via which the I-SIGNAL is received by the imported ECU-INSTANCE, and if the I-SIGNAL's <code>DATA-TYPE-POLICY</code> is set to <code>LEGACY</code>.</p>

Configuration parameters	Mapping description
	<p>ComFilterAlgorithm is set depending on the value of DATA-FILTER-TYPE:</p> <ul style="list-style-type: none"> <li>▶ ALWAYS for ALWAYS.</li> <li>▶ NEVER for NEVER.</li> <li>▶ MASKED_NEW_EQUALS_X for MASKED-NEW-EQUALS-X.</li> <li>▶ MASKED_NEW_DIFFERS_MASKED_OLD for NEW-IS-DIFFERENT.</li> <li>▶ MASKED_NEW_DIFFERS_MASKED_OLD for MASKED-NEW-DIFFERS-MASKED-OLD.</li> <li>▶ MASKED_NEW_DIFFERS_X for MASKED-NEW-DIFFERS-X.</li> <li>▶ NEW_IS_WITHIN for NEW-IS-WITHIN.</li> <li>▶ NEW_IS_OUTSIDE for NEW-IS-OUTSIDE.</li> <li>▶ ONE_EVERY_N for ONE-EVERY-N.</li> </ul> <p>For other DATA-FILTER-TYPE values, ComFilterAlgorithm is not exported.</p> <p>ComFilterMask is set to MASK, if DATA-FILTER-TYPE is one of MASKED-NEW-DIFFERS-MASKED-OLD, MASKED-NEW-DIFFERS-X, MASKED-NEW-EQUALS-X or NEW-IS-DIFFERENT.</p> <p>ComFilterX is set to X, if DATA-FILTER-TYPE is MASKED-NEW-DIFFERS-X or MASKED-NEW-EQUALS-X.</p> <p>ComFilterMax is set to MAX, if DATA-FILTER-TYPE is NEW-IS-OUTSIDE or NEW-IS-WITHIN-X.</p> <p>ComFilterMin is set to MIN, if DATA-FILTER-TYPE is NEW-IS-OUTSIDE or NEW-IS-WITHIN-X.</p> <p>ComFilterOffset is set to OFFSET, if DATA-FILTER-TYPE is ONE-EVERY-N.</p> <p>ComFilterPeriod is set to PERIOD, if DATA-FILTER-TYPE is ONE-EVERY-N.</p>
ComConfig/ComSignal/ComSignalType	<p>In the first step it is determined whether the I-SIGNAL references any DATA-TRANSFORMATION elements via DATA-TRANSFORMATIONS or whether DATA-TYPE-POLICY is set to TRANSFORMING-I-SIGNAL. In these cases, ComSignalType is set to one of the following:</p> <ul style="list-style-type: none"> <li>▶ UINT8_DYN if the I-SIGNAL references a SYSTEM-SIGNAL that has DYNAMIC-LENGTH set to true.</li> <li>▶ UINT8_N if the I-SIGNAL references a SYSTEM-SIGNAL that has DYNAMIC-LENGTH not set or set to false.</li> </ul>

Configuration parameters	Mapping description
	<p>If the I-SIGNAL does not reference any DATA-TRANSFORMATION, the SW-DATA-DEF-PROPS to be used subsequently is retrieved depending on the DATA-TYPE-POLICY of the I-SIGNAL:</p> <ul style="list-style-type: none"> <li>▶ NETWORK-REPRESENTATION-FROM-COM-SPEC: see table "DataTypePolicyEnum" in [1] and specification items [TPS_SYST_02006] and [TPS_SYST_02079] in [3].</li> <li>▶ LEGACY, PORT-INTERFACE-DEFINITION and OVERRIDE: SW-DATA-DEF-PROPS is retrieved via the NETWORK-REPRESENTATION-PROPS of the I-SIGNAL.</li> </ul> <p>In the next step the SW-BASE-TYPE referenced via BASE-TYPE-REF of the SW-DATA-DEF-PROPS is retrieved. ComSignalType is then calculated according to the table <i>SwBaseType to ComSignalType Mapping</i> in [4]. If I-SIGNAL-TYPE of the I-SIGNAL is not available, PRIMITIVE is assumed per default. If BASE-TYPE-SIZE is not available, the I-SIGNAL's LENGTH is used as BASE-TYPE-SIZE. If BASE-TYPE-ENCODING is not available, BOOLEAN is assumed for 1 bit signals and NONE for any other signal bit length. BASE-TYPE-SIZE values are rounded up to 8, 16, 32, or 64 if the BASE-TYPE-ENCODING yields a value of 2C or NONE.</p> <p>If no SW-BASE-TYPE is available directly via BASE-TYPE-REF, the IMPLEMENTATION-DATA-TYPE is retrieved by recursively searching through SW-DATA-DEF-PROPS -&gt; IMPLEMENTATION-DATA-TYPE -&gt; SW-DATA-DEF-PROPS..., until an IMPLEMENTATION-DATA-TYPE with a CATEGORY other than TYPE_REFERENCE is found. If an IMPLEMENTATION-DATA-TYPE's CATEGORY is TYPE_REFERENCE, its SHORT-NAME is boolean, and it references an IMPLEMENTATION-DATA-TYPE with SHORT-NAME set to uint8, ComSignalType is set to BOOLEAN.</p> <p>If no SW-DATA-DEF-PROPS or IMPLEMENTATION-DATA-TYPE was found in the previous steps, ComSignalType is calculated from the I-SIGNAL's LENGTH:</p> <ul style="list-style-type: none"> <li>▶ &gt; 64, or LENGTH not available -&gt; UINT8_N</li> <li>▶ [33 .. 64] -&gt; UINT64</li> <li>▶ [17 .. 32] -&gt; UINT32</li> <li>▶ [9 .. 16] -&gt; UINT16</li> <li>▶ [2 .. 8] -&gt; UINT8</li> <li>▶ 1 -&gt; BOOLEAN</li> </ul>

Configuration parameters	Mapping description
	<p>If the CATEGORY of the found IMPLEMENTATION-DATA-TYPE is ARRAY or STRUCTURE, ComSignalType is set to UINT8_N.</p> <p>If no LOWER-LIMIT/@INTERVAL-TYPE or UPPER-LIMIT/@INTERVAL-TYPE is defined by the DATA-CONSTR referenced via DATA-CONSTR-REF, ComSignalType is calculated using the SW-BASE-TYPE referenced by the IMPLEMENTATION-DATA-TYPE, or the I-SIGNAL's LENGTH if the SW-BASE-TYPE is not available.</p> <p>If either LOWER-LIMIT/@INTERVAL-TYPE or UPPER-LIMIT/@INTERVAL-TYPE is INFINITE, ComSignalType is set to FLOAT32 or FLOAT64 (I-SIGNAL's LENGTH &gt; 32).</p> <p>If the BASE-TYPE-ENCODING as defined by the SW-BASE-TYPE referenced by the IMPLEMENTATION-DATA-TYPE via BASE-TYPE-REF equals IEEE754, ComSignalType is set to FLOAT32 or FLOAT64 (I-SIGNAL's LENGTH &gt; 32).</p> <p>LOWER-LIMIT/UPPER-LIMIT elements having INTERVAL-TYPE elements which are OPEN are converted to CLOSED first, i.e. ]-1 .. 8[ is converted to [0 .. 7]. Then ComSignalType is calculated from the bit length required to cover the value range LOWER-LIMIT/UPPER-LIMIT according to the lists below. Note that the BASE-TYPE-SIZE of a referenced SW-BASE-TYPE overrides this calculated bit length. If BASE-TYPE-ENCODING is available as well, ComSignalType is calculated according to the table "SwBaseType to ComSignalType Mapping" in [1].</p> <p>Bit length of UPPER-LIMIT (value of converted LOWER-LIMIT &gt;= 0):</p> <ul style="list-style-type: none"> <li>▶ &gt; 64 -&gt; UINT8_N</li> <li>▶ [33 .. 64] -&gt; UINT64</li> <li>▶ [17 .. 32] -&gt; UINT32</li> <li>▶ [9 .. 16] -&gt; UINT16</li> <li>▶ [2 .. 8] -&gt; UINT8</li> <li>▶ 1 -&gt; BOOLEAN</li> </ul> <p>Bit length of LOWER-LIMIT/UPPER-LIMIT (whichever is greater, value of converted LOWER-LIMIT &lt; 0):</p> <ul style="list-style-type: none"> <li>▶ &gt; 64 -&gt; UINT8_N</li> <li>▶ [33 .. 64] -&gt; SINT64</li> </ul>



Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ [17 .. 32] -&gt; SINT32</li> <li>▶ [9 .. 16] -&gt; SINT16</li> <li>▶ [2 .. 8] -&gt; SINT8</li> <li>▶ 1 -&gt; BOOLEAN</li> </ul>
ComConfig/ComSignalGroup	<p>For every signal instance which represents an I-SIGNAL-GROUP and which is sent or received by the imported ECU-INSTANCE, a ComSignalGroup container is created. The container name is GR&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-GROUP.</p> <p>For the setting of ComTimeout, ComFirstTimeout, ComDataInvalidAction, and ComUpdateBitPosition, see ComConfig/ComSignal.</p> <p>If the I-SIGNAL-GROUP is contained in an NM-PDU, ComTransferProperty is set to PENDING.</p> <p>If the I-SIGNAL-GROUP is contained in an I-SIGNAL-I-PDU, the algorithm collects all distinct and valid I-SIGNAL-TO-I-PDU-MAPPING/TRANSFER-PROPERTY values of all I-SIGNAL-TO-I-PDU-MAPPING elements that are associated with any I-SIGNAL element of the I-SIGNAL-GROUP. If the collection is empty, the I-SIGNAL-TO-I-PDU-MAPPING/TRANSFER-PROPERTY value of the I-SIGNAL-GROUP is used to configure the ComTransferProperty according to ComConfig/ComSignal.</p> <p>If the collection contains exactly one value, this TRANSFER-PROPERTY is used to configure the ComTransferProperty.</p> <p>If the collection contains two values and one of them is PENDING, the other value is used to configure the ComTransferProperty. In any other case an error is issued.</p> <p>ComSystemTemplateSignalGroupRef is set to the AUTOSAR path of the I-SIGNAL-TO-I-PDU-MAPPING of the signal instance.</p> <p>If the I-SIGNAL-GROUP references a DATA-TRANSFORMATION via COM-BASED-SIGNAL-GROUP-TRANSFORMATIONS/DATA-TRANSFORMATION-REF-CONDITIONAL/DATA-TRANSFORMATION-REF, ComSignalGroupArrayAccess is set to true, otherwise ComSignalGroupArrayAccess is set to false.</p> <p>The ComSignalGroupDirection is set to SEND if the PDU instance is sent by the imported ECU-INSTANCE. Otherwise it is set to RECEIVE.</p>

Configuration parameters	Mapping description
ComConfig/ComSignalGroup/ComGroupSignal	<p>For every SYSTEM-SIGNAL which is part of the I-SIGNAL-GROUP ("GroupSignal"), a ComGroupSignal container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the SYSTEM-SIGNAL.</p> <p>PACKING-BYTE-ORDER of the GroupSignal's I-SIGNAL-TO-I-PDU-MAPPING must be the same for all GroupSignals of the I-SIGNAL-GROUP. Different PACKING-BYTE-ORDER values result in an error.</p> <p>For the setting of ComSignalInitValue, ComSignalDataInvalidValue, ComBitPosition, ComSignalLength, ComSignalEndianness, ComBitSize and ComSignalType see ComConfig/ComSignal.</p> <p>ComSystemTemplateSystemSignalRef is set to the AUTOSAR path of the I-SIGNAL-TO-I-PDU-MAPPING of the signal instance of the GroupSignal.</p> <p>ComFilter is set only for GroupSignals which are sent by the imported ECU-INSTANCE, see ComSignal/ComFilter.</p> <p>ComTransferProperty is set depending on the value of the GroupSignal's I-SIGNAL-TO-I-PDU-MAPPING/TRANSFER-PROPERTY:</p> <ul style="list-style-type: none"> <li>▶ PENDING for PENDING.</li> <li>▶ TRIGGERED_ON_CHANGE for TRIGGERED-ON-CHANGE.</li> </ul> <p>For all other values a warning is issued and ComTransferProperty is not set.</p>
ComConfig/ComIPduGroup	<p>For every I-SIGNAL-I-PDU-GROUP associated with the imported ECU-INSTANCE, a ComIPduGroup container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-I-PDU-GROUP. For I-SIGNAL-I-PDU-GROUP elements contained within other I-SIGNAL-I-PDU-GROUP elements references to the containers of the parent groups are set in the reference list ComIPduGroupGroupRef.</p> <p>If no I-SIGNAL-I-PDU-GROUP element exists that references an I-SIGNAL-I-PDU which is sent or received by the imported ECU-INSTANCE, two default ComIPduGroup elements are created, named RXPBUS_GLOBAL and TXPBUS_GLOBAL. Then for each CAN-CLUSTER, FLEXRAY-CLUSTER, LIN-CLUSTER and ETHERNET-PHYSICAL-CHANNEL to which the imported ECU-INSTANCE is connected, two ComIPduGroup elements are created named RG&lt;SHORT-NAME of CLUSTER or PHYSICAL-CHANNEL&gt; and TG&lt;SHORT-NAME of CLUSTER or PHYSICAL-CHANNEL&gt;. Via ComIPduGroupGroupRef a reference to the parent ComIPduGroup is set, RXPBUS_GLOBAL for all RG groups, TXPBUS_GLOBAL for all TG groups.</p>

Configuration parameters	Mapping description
	<p>If no I-SIGNAL-I-PDU-GROUP element exists that references an NM-PDU which is sent or received by the imported ECU-INSTANCE, two default ComIPduGroup elements are created, named RXNMPDUS_GLOBAL and TXNMPDUS_GLOBAL. Then for each CAN-CLUSTER, FLEXRAY-CLUSTER and ETHERNET-PHYSICAL-CHANNEL to which the imported ECU-INSTANCE is connected, two ComIPduGroup elements are created named RNG&lt;SHORT-NAME of COMMUNICATION-CLUSTER&gt; and TNG&lt;SHORT-NAME of COMMUNICATION-CLUSTER&gt;. Via ComIPduGroupGroupRef a reference to the parent ComIPduGroup is set, RXNMPDUS_GLOBAL for all RG groups, TXNMPDUS_GLOBAL for all TG groups. If no NM-PDU elements exist for one direction (Rx/Tx), no dedicated ComIPduGroup is created for this direction.</p>
ComConfig/ComTimeBase	<p>ComConfigurationGwTimeBase is set to COM-CONFIGURATION-GW-TIME-BASE.</p> <p>ComConfigurationRxTimeBase is set to COM-CONFIGURATION-RX-TIME-BASE.</p> <p>ComConfigurationTxTimeBase is set to COM-CONFIGURATION-TX-TIME-BASE.</p>
ComConfig/ComGwMapping	<p>For each received signal instance representing an I-SIGNAL or an I-SIGNAL-GROUP, which is referenced via a SOURCE-SIGNAL-REF -&gt; I-SIGNAL-TRIGGERING that belongs to an I-SIGNAL-MAPPING in the GATEWAY of the imported ECU-INSTANCE, a ComGwMapping container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL or the I-SIGNAL-GROUP.</p>
ComConfig/ComGwMapping/ComGwSource	<p>The choice container is always set to ComGwSignal.</p> <p>ComGwSignal/ComGwSignalRef references the corresponding ComSignal or ComSignalGroup container created for the signal instance used to create the ComGwMapping container.</p>
ComConfig/ComGwMapping/ComGwDestination	<p>For each sent signal instance representing an I-SIGNAL or an I-SIGNAL-GROUP, which is referenced via a TARGET-SIGNAL-REF -&gt; I-SIGNAL-TRIGGERING that belongs to an I-SIGNAL-MAPPING and, which also references the I-SIGNAL or I-SIGNAL-GROUP of the ComGwSource, a ComGwDestination container is created. The container name is GM&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL or the I-SIGNAL-GROUP.</p> <p>ComGwSignal/ComGwSignalRef references the corresponding ComSignal or ComSignalGroup container.</p>

### 3.4.8.1. Configuration for partial networking ERA and EIRA PDUs and signals

Configuration parameters	Mapping description
ComConfig/ComIPdu	<p>For every ERA and EIRA <code>Pdu</code> container created for PNC-enabled NM-CLUSTER elements in the <code>EcuC</code> module (see <a href="#">Section 3.4.14, “EcuC”</a>), a <code>ComIPdu</code> is created. The container name equals the name of the <code>Pdu</code> container in the <code>EcuC</code> module.</p> <p>The single <code>ComIPduSignalRef</code> entry references the <code>ComSignal</code> container also created for this PDU.</p> <p><code>ComIPduDirection</code> is set to <code>RECEIVE</code>.</p> <p>The single <code>ComIPduGroupRef</code> entry references the <code>ComIPduGroup</code> also created for this PDU.</p> <p><code>ComPduIdRef</code> references the <code>Pdu</code> container in the <code>EcuC</code> module.</p>
ComConfig/ComSignal	<p>For every created <code>ComIPdu</code> container, a corresponding <code>ComSignal</code> container is created. The container name is <code>SG&lt;ComIPdu name&gt;</code>.</p> <p><code>ComBitSize</code> is set to <code>PNC-VECTOR-LENGTH * 8</code>.</p> <p><code>ComSignalLength</code> is set to <code>PNC-VECTOR-LENGTH</code>.</p> <p><code>ComSignalInitValue</code> is set to 0, converted to a space separated byte array where <code>PNC-VECTOR-LENGTH</code> defines the length of this byte array.</p> <p>If <code>PNC-VECTOR-LENGTH</code> is not defined, a warning is issued for <code>ComBitSize</code>, <code>ComSignalLength</code>, and <code>ComSignalInitValue</code> and these parameters are not set.</p> <p><code>ComBitPosition</code> is set to 0.</p> <p><code>ComSignalDirection</code> is set to <code>SEND</code> if the PDU instance is sent by the imported ECU-INSTANCE. Otherwise it is set to <code>RECEIVE</code>.</p> <p><code>ComSignalEndianness</code> is set to <code>OPAQUE</code>.</p> <p><code>ComSignalType</code> is set to <code>UINT8_N</code>.</p> <p><code>ComNotification</code> is set to <code>ComM_COMCbk_&lt;ComSignal name&gt;</code>.</p>
ComConfig/ComIPduGroup	<p>For every created EIRA <code>ComIPdu</code> container, a corresponding <code>ComIPduGroup</code> container is created. The container name depends on the <code>EcuC Pdu</code> container for which the <code>ComIPdu</code> has been created:</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ <b>RNGEIRACanNm</b> for the <b>CanNmPnEiraRxNSdu</b> container. It references the global <b>ComIPduGroup</b> named <b>RXEIRAPDUS_GLOBAL</b>.</li> <li>▶ <b>RNGEIRAFrNm</b> for the <b>FlexRayNmPnEiraRxNSdu</b> container. It references the global <b>ComIPduGroup</b> named <b>RXEIRAPDUS_GLOBAL</b>.</li> <li>▶ <b>RNGEIRAUdpNm</b> for the <b>UdpNmPnEiraRxNSdu</b> container. It references the global <b>ComIPduGroup</b> named <b>RXEIRAPDUS_GLOBAL</b>.</li> </ul> <p>For every created ERA <b>ComIPdu</b> container, a corresponding <b>ComIPduGroup</b> container is created. The container name is <b>RNG&lt;name&gt;</b> for received PDUs, and <b>TNG&lt;name&gt;</b> for sent PDUs, where <b>&lt;name&gt;</b> is the mangled name of the <b>CAN-COMMUNICATION-CLUSTER</b>, <b>FLEXRAY-COMMUNICATION-CLUSTER</b>, or <b>ETHERNET-PHYSICAL-CHANNEL</b> on which the PDU is sent or received. The referenced parent <b>ComIPduGroup</b> is <b>RXNMPDUS_GLOBAL</b> for <b>ComIPduGroup</b> containers of received PDUs and <b>TXNMPDUS_GLOBAL</b> for <b>ComIPduGroup</b> containers of sent PDUs.</p>

### 3.4.8.2. Configuration for partial networking IRA signals

An *IRA* (internal request array) signal is represented by a **ComSignal** container that is associated with a **ComIPdu** container. This **ComIPdu** container has been created for an **NmUserDataPdu**, which the ECU-INSTANCE is sending in a PNC-enabled NM-CLUSTER. The bit area of an IRA signal covers the PNC vector which is defined by the **PNC-VECTOR-OFFSET** and **PNC-VECTOR-LENGTH** of the SYSTEM. If the NM-PDU that represents the **NmUserDataPdu** contains an **I-SIGNAL**, which completely covers the PNC vector, the signal is picked for IRA signal configuration.

If the NM-PDU does not contain such an **I-SIGNAL**, a dedicated **ComSignal** container is created. The container name is **<name>NmPnIraTxNSdu**, where **<name>** is the mangled name of the **CAN-COMMUNICATION-CLUSTER**, **FLEXRAY-COMMUNICATION-CLUSTER**, or **ETHERNET-PHYSICAL-CHANNEL** on which the IRA signal is sent.

The following **ComSignal** parameters for IRA signals are configured as described for EIRA and ERA signals in [Section 3.4.8.1, “Configuration for partial networking ERA and EIRA PDUs and signals”](#):

- ▶ **ComBitSize**
- ▶ **ComSignalDirection**
- ▶ **ComSignalLength**
- ▶ **ComSignalEndianness**
- ▶ **ComSignalType**

ComBitOffset is set to  $(\text{PNC-VECTOR-OFFSET} - \text{<non user data area>}) * 8$ , where <non user data area> is the byte length of the non user data area of the NM-PDU as described in [Section 3.4.14, “EcuC”](#).

### 3.4.9. ComM

Configuration parameters	Mapping description
ComMGeneral	ComMPncGatewayEnabled is set to false if all COMMUNICATION-CONNECTOR elements of the imported ECU-INSTANCE have the parameter PNC-GATEWAY-TYPE set to NONE. Otherwise ComMPncGatewayEnabled is set to true. If PNC-GATEWAY-TYPE is undefined for any of the COMMUNICATION-CONNECTOR elements, ComMPncGatewayEnabled is not set and a warning is reported. ComMPncPrepareSleepTimer is set to PNC-PREPARE-SLEEP-TIMER.
ComMConfigSet	ComMPncEnabled is set to true if any PNC-enabled NM-CLUSTER (see <a href="#">Section 3.4.14, “EcuC”</a> ) exists. Otherwise ComMPncEnabled is not set.
ComMConfigSet/ComMUser	<p>One ComMUser container is created for each COM-MGR-USER-NEEDS element for which the following conditions hold</p> <ul style="list-style-type: none"> <li>▶ The aggregating SWC-SERVICE-DEPENDENCY refers via REPRESENTED-PORT-GROUP-REF to a valid PORT-GROUP</li> <li>▶ One or more COM-MANAGEMENT-MAPPING elements refer to that PORT-GROUP as well</li> <li>▶ The COM-MANAGEMENT-MAPPING elements refer to one or more CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, LIN-CLUSTER, or ETHERNET-PHYSICAL-CHANNEL elements to which the imported ECU-INSTANCE is connected. The COM-MANAGEMENT-MAPPING refers to these elements either directly via COM-MANAGEMENT-PORT-GROUP-IREFS/COM-MANAGEMENT-PORT-GROUP-IREF/TARGET-REF, or indirectly via COM-MANAGEMENT-GROUP-REFS/COM-MANAGEMENT-GROUP-REF. In the latter case, all referenced I-SIGNAL-I-PDU-GROUP elements and their sub groups are retrieved. In the second step, all PDUs belonging to the I-SIGNAL-I-PDU-GROUP elements are collected. The set of CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, LIN-CLUSTER, or ETHERNET-PHYSICAL-CHANNEL elements via which the imported ECU-INSTANCE sends or receives any of the PDUs are considered the elements indirectly referenced via COM-MANAGEMENT-GROUP-REFS/COM-MANAGEMENT-GROUP-REF.</li> </ul> <p>The ComMUser container is related to all ComMChannel containers that are set up for the CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, LIN-</p>

Configuration parameters	Mapping description
	<p>CLUSTER, or ETHERNET-PHYSICAL-CHANNEL elements of the COM-MANAGEMENT-MAPPING elements. The name of the ComMUser container is the mangled name of the COM-MGR-USER-NEEDS element.</p>
ComMConfigSet/ComMChannel	<p>For each CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, LIN-CLUSTER and ETHERNET-PHYSICAL-CHANNEL to which the imported ECU-INSTANCE is connected, a ComMChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COMMUNICATION-CLUSTER or ETHERNET-PHYSICAL-CHANNEL.</p> <p>A ComMChannel is considered to be a <i>managing</i> channel in the following cases</p> <ul style="list-style-type: none"> <li>▶ The ComMChannel is represented by an ETHERNET-PHYSICAL-CHANNEL and that ETHERNET-PHYSICAL-CHANNEL refers via MANAGED-PHYSICAL-CHANNEL-REF to one or more ETHERNET-PHYSICAL-CHANNEL elements or to PHYSICAL-CHANNEL elements that are contained in CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, or LIN-CLUSTER elements which in turn represent the <i>managed</i> ComMChannel elements.</li> <li>▶ The ComMChannel is represented by a CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, or LIN-CLUSTER and one or more of the PHYSICAL-CHANNEL elements of the cluster refer via MANAGED-PHYSICAL-CHANNEL-REF to one or more ETHERNET-PHYSICAL-CHANNEL elements or to PHYSICAL-CHANNEL elements that are contained in CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, or LIN-CLUSTER elements which in turn represent the <i>managed</i> ComMChannel elements.</li> </ul> <p>ComMBusType is set depending on the type of the COMMUNICATION-CLUSTER or ETHERNET-PHYSICAL-CHANNEL:</p> <ul style="list-style-type: none"> <li>▶ COMM_BUS_TYPE_CAN for CAN-CLUSTER.</li> <li>▶ COMM_BUS_TYPE_CAN for J-1939-CLUSTER.</li> <li>▶ COMM_BUS_TYPE_FR for FLEXRAY-CLUSTER.</li> <li>▶ COMM_BUS_TYPE_ETH for ETHERNET-PHYSICAL-CHANNEL.</li> <li>▶ COMM_BUS_TYPE_LIN for LIN-CLUSTER.</li> <li>▶ For any other cluster type ComMBusType is not set and a warning is issued.</li> </ul> <p>ComMPncGatewayType is not configured if the ComMChannel is a <i>managed</i> ComMChannel. Otherwise ComMPncGatewayType is set depending on the PNC-GATEWAY-TYPE of the COMMUNICATION-CONNECTOR that connects</p>

Configuration parameters	Mapping description
	<p>the imported ECU-INSTANCE to the COMMUNICATION-CLUSTER or ETHER-NET-PHYSICAL-CHANNEL:</p> <ul style="list-style-type: none"> <li>▶ COMM_GATEWAY_TYPE_ACTIVE for ACTIVE</li> <li>▶ COMM_GATEWAY_TYPE_PASSIVE for PASSIVE</li> </ul> <p>ComMPncGatewayType is not configured if PNC-GATEWAY-TYPE is not defined or set to NONE.</p> <p>If the ComMChannel is a <i>managed</i> ComMChannel, ComMManageReference is configured to refer to the corresponding <i>managing</i> ComMChannel container.</p> <p>If any ComMUser container has been created which is associated with this ComMChannel container, one ComMUserPerChannel container is created. The name of the container is the same as the name of the ComMUser container. The reference ComMUserChannel is configured to refer to the ComMUser container.</p> <p>ComMNetworkManagement/ComMNmVariant is set to LIGHT for <i>managed</i> ComMChannel containers, and to FULL for <i>managing</i> ComMChannel containers. If the ComMChannel is neither <i>managed</i> nor <i>managing</i> the following rules apply.</p> <ul style="list-style-type: none"> <li>▶ For LIN-CLUSTER elements, ComMNmVariant is set to LIGHT</li> <li>▶ If no NM-NODE exists that connects an NM-CLUSTER to the COMMUNICATION-CLUSTER, or if the NM-NODE does not reference any sent or received NM-PDU elements, ComMNmVariant is set to NONE unless ComMNmVariant has been manually configured as LIGHT, in which case that value is left unchanged.</li> <li>▶ If an NM-NODE exists that connects a NM-CLUSTER to the COMMUNICATION-CLUSTER, and if the NM-NODE references at least one sent or received NM-PDU element, ComMNmVariant is set to PASSIVE if the NM-NODE's NM-PASSIVE-MODE-ENABLED is set to true. If NM-PASSIVE-MODE-ENABLED is not set or set to false, ComMNmVariant is set to FULL.</li> </ul>
ComMConfigSet/ComMPnc	<p>For every PNC-MAPPING for which a valid and unique Id can be calculated and which fulfills at least one of the following conditions:</p> <ul style="list-style-type: none"> <li>▶ The PNC-MAPPING is referencing an I-SIGNAL-I-PDU-GROUP which is also referenced by the imported ECU-INSTANCE via ASSOCIATED-COM-I-PDU-GROUP-REFS</li> <li>▶ The PNC-MAPPING is referencing at least one PHYSICAL-CHANNEL to which the configured ECU is connected as well</li> </ul>



Configuration parameters	Mapping description
	<p>, a ComMPnc container is created. Each ComMPnc is identified by its Id. The formula for calculating this Id depends on the AUTOSAR specification version that the ComM module implements.</p> <p>For ComM modules that implement AUTOSAR 4.1.0 or lower, the formula <math>Id := PNC-IDENTIFIER - (PNC-VECTOR-OFFSET * 8)</math> is applied, where PNC-VECTOR-OFFSET is defined at the SYSTEM to which the PNC-IDENTIFIER belongs. For ComM modules that implement AUTOSAR 4.1.1 or higher, <math>Id := PNC-IDENTIFIER</math> is applied.</p> <p>The container name of the ComMPnc is ComMPnc_&lt;Id&gt;. If Id cannot be calculated due to a missing PNC-IDENTIFIER or PNC-VECTOR-OFFSET, a warning is issued and no ComMPnc container is created.</p> <p>If multiple PNC-MAPPING elements exist that have the same PNC-IDENTIFIER, a warning is issued and only the first PNC-MAPPING is used.</p> <p>ComMPncId is set to Id.</p> <p>For every COMMUNICATION-CLUSTER or ETHERNET-PHYSICAL-CHANNEL through which either any of the I-SIGNAL-I-PDU-GROUP's PDU elements are sent, and/or received, and/or routed, or which is referenced via PNC-MAPPING/PHYSICAL-CHANNEL-REF, one ComMChannelPerPnc reference is created. This ComMChannelPerPnc references the ComMChannel that has been created for the COMMUNICATION-CLUSTER or ETHERNET-PHYSICAL-CHANNEL.</p> <p>If an EthIfSwitchPortGroup container has been created in the EthIf module for the PNC-MAPPING, ComMPncEthIfSwitchPortGroupRef is configured to refer to this container. <a href="#">Section 3.4.17, "EthIf"</a> describes the configuration of EthIfSwitchPortGroup containers in detail.</p>
ComMConfigSet/ComMPnc/ComMPncComSignal	<p>If a PNC-enabled CAN-NM-CLUSTER that sends/receives a PDU of the PNC-MAPPING exists, one ComMPncComSignal container named CanNmPnEiraRxNSdu is created.</p> <p>If a PNC-enabled UDP-NM-CLUSTER that sends/receives a PDU of the PNC-MAPPING exists, one ComMPncComSignal container named UdpNmPnEiraRxNSdu is created.</p> <p>If a PNC-enabled FLEXRAY-NM-CLUSTER that sends/receives a PDU of the PNC-MAPPING exists, one ComMPncComSignal container named FlexRayNmPnEiraRxNSdu is created.</p>

Configuration parameters	Mapping description
	<p>For every <code>ComSignal</code> container that has been set up in <code>Com</code> for an EIRA, ERA, or IRA signal that belongs to this <code>ComMPnc</code> container, one <code>ComMPncComSignal</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the signal. The creation of <code>ComSignal</code> containers for EIRA, ERA, and IRA signals is described in <a href="#">Section 3.4.8.1, “Configuration for partial networking ERA and EIRA PDUs and signals”</a> and in <a href="#">Section 3.4.8.2, “Configuration for partial networking IRA signals”</a>. For detailed information about the PNC Vector, see <a href="#">Section 3.4.4, “CanNm”</a>.</p> <p>The <i>IRA signal kind</i> of an IRA signals depends on the <code>PNC-GATEWAY-TYPE</code> of the <code>COMMUNICATION-CONNECTOR</code> that connects the imported <code>ECU-INSTANCE</code> to the <code>PHYSICAL-CHANNEL</code> on which the signal is sent. If the <code>PNC-GATEWAY-TYPE</code> is set to <code>PASSIVE</code>, the <i>IRA signal kind</i> of the IRA signal is ERA. For all other values, its <i>IRA signal kind</i> is EIRA.</p> <p>The following parameters of <code>ComMPncComSignal</code> are set:</p> <ul style="list-style-type: none"> <li>▶ <code>ComMPncComSignalDirection</code> is set to <code>RX</code> for ERA and EIRA signals, and to <code>TX</code> for IRA signals.</li> <li>▶ <code>ComMPncComSignalKind</code> is set to <code>EIRA</code> for EIRA and to <code>ERA</code> for ERA signals. For IRA signals, the <i>IRA signal kind</i> is configured.</li> <li>▶ <code>ComMPncComSignalRef</code> references the corresponding <code>ComSignal</code> container in the <code>Com</code> module.</li> <li>▶ For ERA signals and for IRA signals having an <i>IRA signal kind</i> of ERA, <code>ComMPncComSignalChannelRef</code> references the <code>ComMChannel</code> container that has been created for the PNC-enabled <code>NM-CLUSTER</code>.</li> </ul>

### 3.4.10. Csm

Several client BSW modules are built onto of the cryptographic services which the `Csm` BSW module is providing. Since each of these client BSW modules requires a differently configured `Csm` infrastructure, this chapter describes the configuration process for each client in a dedicated sub chapter.

### 3.4.10.1. Csm for SecOC

Configuration parameters	Mapping description
Csm/CsmJobs/CsmJob	<p>For every SECURED-I-PDU the configured ECU-INSTANCE is sending or receiving, the associated PDU-TRIGGERING is retrieved. If PDU-TRIGGERING/SEC-OC-CRYPTO-MAPPING-REF refers to a valid SEC-OC-CRYPTO-SERVICE-MAPPING element, one CsmJob container is created. The container name is &lt;Primitive&gt;_&lt;PduInstName&gt;, where &lt;PduInstName&gt; is the name of the PDU instance, and &lt;Primitive&gt; is CsmMacVerify for received PDU instances and CsmMacGenerate for sent PDU instances.</p> <p>CsmJobKeyRef is set to reference the container that has been created for the CRYPTO-SERVICE-KEY referenced via SEC-OC-CRYPTO-SERVICE-MAPPING/CRYPTO-SERVICE-KEY-REF.</p> <p>CsmJobPrimitiveRef is set to reference the container that has been created for the CRYPTO-SERVICE-PRIMITIVE referenced via SEC-OC-CRYPTO-SERVICE-MAPPING/AUTHENTICATION-REF.</p> <p>CsmJobQueueRef is set to reference the container that has been created for the CRYPTO-SERVICE-QUEUE referenced via SEC-OC-CRYPTO-SERVICE-MAPPING/CRYPTO-SERVICE-QUEUE-REF.</p>
Csm/CsmQueues/CsmQueue	<p>For every CRYPTO-SERVICE-QUEUE which is referenced by a SEC-OC-CRYPTO-SERVICE-MAPPING of a SECURED-I-PDU the configured ECU-INSTANCE is sending or receiving, one CsmQueue container is created. The container name is the SHORT-NAME of the CRYPTO-SERVICE-QUEUE.</p> <p>CsmQueueSize is set to CRYPTO-SERVICE-QUEUE/QUEUE-SIZE.</p>
Csm/CsmKeys/CsmKey	<p>For every CRYPTO-SERVICE-KEY which is referenced by a SEC-OC-CRYPTO-SERVICE-MAPPING of a SECURED-I-PDU the configured ECU-INSTANCE is sending or receiving, one CsmKey container is created. The container name is the SHORT-NAME of the CRYPTO-SERVICE-KEY.</p>
Csm/CsmPrimitives	<p>For every CRYPTO-SERVICE-PRIMITIVE which is referenced by a SEC-OC-CRYPTO-SERVICE-MAPPING of a SECURED-I-PDU the configured ECU-INSTANCE is sending or receiving, one CsmPrimitives container is created. The container name is &lt;Primitive&gt;_&lt;PduInstName&gt;, where &lt;PduInstName&gt; is the name of the PDU instance, and &lt;Primitive&gt; is CsmMacVerify for received PDUs and CsmMacGenerate for sent PDUs.</p>
Csm/CsmPrimitives/CsmMacVerify/CsmMacVerifyConfig	<p>If the SECURED-I-PDU is received, one Csm/CsmPrimitives/CsmMacVerify/CsmMacVerifyConfig container is created, the container name is set to</p>

Configuration parameters	Mapping description
fy/CsmMacVerifyConfig	<p>CsmMacVerify_&lt;PduInstName&gt;, where &lt;PduInstName&gt; is the name of the PDU instance.</p> <p>CsmMacVerifyAlgorithmFamily is set to CRYPTO-SERVICE-PRIMITIVE/ALGORITHM-FAMILY unless CRYPTO-SERVICE-PRIMITIVE/ALGORITHM-FAMILY is either CRYPTO_ALGOFAM_SHAKE128 or CRYPTO_ALGOFAM_SHAKE256. in the former case, CRYPTO_ALGOFAM_SHA3_SHAKE128 is configured, in the latter case, CRYPTO_ALGOFAM_SHA3_SHAKE256 is configured.</p> <p>CsmMacVerifyAlgorithmKeyLength is set to CRYPTO-SERVICE-KEY/LENGTH of the CRYPTO-SERVICE-KEY referenced via SEC-OC-CRYPTO-SERVICE-MAPPING/CRYPTO-SERVICE-KEY-REF.</p> <p>CsmMacVerifyAlgorithmSecondaryFamily is set to CRYPTO_ALGOFAM_CUSTOM if CRYPTO-SERVICE-PRIMITIVE/ALGORITHM-SECONDARY-FAMILY is configured.</p> <p>CsmMacVerifyAlgorithmSecondaryFamilyCustom is set to CRYPTO-SERVICE-PRIMITIVE/ALGORITHM-SECONDARY-FAMILY.</p>
Csm/CsmPrimitives/CsmMacGenerate/CsmMacGenerateConfig	<p>If the SECURED-I-PDU is sent, one Csm/CsmPrimitives/CsmMacGenerate/CsmMacGenerateConfig container is created, the container name is set to CsmMacGenerate_&lt;PduInstName&gt;, where &lt;PduInstName&gt; is the name of the PDU instance.</p> <p>CsmMacGenerateAlgorithmFamily is configured in the same way as CsmMacVerifyAlgorithmFamily for received SECURED-I-PDU elements.</p> <p>CsmMacGenerateAlgorithmKeyLength is configured in the same way as CsmMacVerifyAlgorithmKeyLength for received SECURED-I-PDU elements.</p> <p>CsmMacGenerateAlgorithmSecondaryFamily is configured in the same way as CsmMacVerifyAlgorithmSecondaryFamily for received SECURED-I-PDU elements.</p> <p>CsmMacGenerateAlgorithmSecondaryFamilyCustom is configured in the same way as CsmMacVerifyAlgorithmSecondaryFamilyCustom for received SECURED-I-PDU elements.</p>

### 3.4.10.2. Csm for Tls

Configuration containers are configured for every `TLS-CRYPTO-SERVICE-MAPPING` for which a `TlsConnection` container was created and a valid cipher suite could be determined as described in [Section 3.4.47, “Tls”](#).

For each `TlsConnection` container, `CsmJob` containers are created for

- ▶ `PRFPreMasterSecret`
- ▶ `PRFMasterSecret`
- ▶ `MACGenerate`
- ▶ `MACVerify`
- ▶ `Hash`
- ▶ `Random`

If the cipher suite supports encryption, additional jobs are created for

- ▶ `Encrypt`
- ▶ `Decrypt`

If the cipher suite uses key exchange, additional job is created for `SignatureGenerate`.

The names of the `CsmJob` is `<Prefix>_<SocketAddress>`, where `<Prefix>` corresponds to the job type described above and `<SocketAddress>` is the name of the `SOCKET-ADDRESS` associated with the `TlsConnection`.

For each of the created `CsmJob` containers, one `CsmKey` and one `CsmPrimitive` container are created. The container names equal the name for the `CsmJob` container. The `CsmJob` container references the `CsmKey` container via `CsmJobKeyRef` and the `CsmPrimitive` container via `CsmJobPrimitiveRef`.

For `PRFPreMasterSecret` `CsmPrimitive` containers a `CsmMacGenerate` sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
<code>CsmMacGenerateAlgorithmFamily</code>	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.g. <code>CRYPTO_ALGOFAM_SHA2_256</code> for <code>TLS_PSK_WITH_AES_128_GCM_SHA256</code> .
<code>CsmMacGenerateAlgorithmKeyLength</code>	For PSK cipher suites: $2 * \text{<PSKLen>} + 4$ , where <code>&lt;PSKLen&gt;</code> is the byte length of the pre-shared key.
<code>CsmMacGenerateAlgorithmMode</code>	<code>CRYPTO_ALGOMODE_HMAC</code>
<code>CsmMacGenerateResultLength</code>	Byte length of the hash function of the cipher suite

Module Configuration Parameter	Value
CsmMacGenerateDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.

For PRFMasterSecret CsmPrimitive containers a CsmMacGenerate sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmMacGenerateAlgorithmFamily	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.-g. CRYPTO_ALGOFAM_SHA2_256 for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmMacGenerateAlgorithmKeyLength	48
CsmMacGenerateAlgorithmMode	CRYPTO_ALGOMODE_HMAC
CsmMacGenerateResultLength	Byte length of the hash function of the cipher suite
CsmMacGenerateDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.

For MACGenerate CsmPrimitive containers a CsmMacGenerate sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmMacGenerateAlgorithmFamily	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.-g. CRYPTO_ALGOFAM_SHA2_256 for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmMacGenerateAlgorithmKeyLength	Byte length of the hash function of the cipher suite
CsmMacGenerateAlgorithmMode	CRYPTO_ALGOMODE_HMAC
CsmMacGenerateResultLength	Byte length of the hash function of the cipher suite
CsmMacGenerateDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.

For MACVerify CsmPrimitive containers a CsmMacVerify sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmMacVerifyAlgorithmFamiliy	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.-g. CRYPTO_ALGOFAM_SHA2_256 for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmMacVerifyAlgorithmKeyLength	Byte length of the hash function of the cipher suite
CsmMacVerifyAlgorithmMode	CRYPTO_ALGOMODE_HMAC
CsmMacVerifyCompareLength	Byte length of the hash function of the cipher suite
CsmMacVerifyDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.

For Hash CsmPrimitive containers a CsmHash sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmHashAlgorithmFamiliy	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.-g. CRYPTO_ALGOFAM_SHA2_256 for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmHashResultLength	Byte length of the hash function of the cipher suite
CsmHashDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.

For Random CsmPrimitive containers a CsmRandomGenerate sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmRandomGenerateAlgorithmFamiliy	CRYPTO_ALGOFAM_RNG
CsmRandomGenerateResultLength	Byte length of the hash function of the cipher suite

For Encrypt CsmPrimitive containers a CsmAEADEncrypt sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmAEADEncryptAlgorithmFamiliy	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.g.

Module Configuration Parameter	Value
	CRYPTO_ALGOFAM_AES for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmAEADEncryptAlgorithmKeyLength	Byte length of the key of the cipher suite's encryption function
CsmAEADEncryptAlgorithmMode	Enumeration value that corresponds to the mode of the encryption algorithm of the cipher suite, e.-g. CRYPTO_ALGOMODE_GCM for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmAEADEncryptTagLength	Byte length of the key of the cipher suite's encryption function
CsmAEADEncryptAssociatedDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.
CsmAEADEncryptCiphertextMaxLength	Same as CsmAEADEncryptAssociatedDataMaxLength
CsmAEADEncryptPlaintextMaxLength	Same as CsmAEADEncryptAssociatedDataMaxLength
CsmAEADEncryptKeyRef	Refers to the CsmKey container the CsmJob references via CsmJobKeyRef.

For Decrypt CsmPrimitive containers a CsmAEADDecrypt sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmAEADDecryptAlgorithmFamiliy	Enumeration value that corresponds to the hash algorithm and hash length of the cipher suite, e.g. CRYPTO_ALGOFAM_AES for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmAEADDecryptAlgorithmKeyLength	Byte length of the key of the cipher suite's encryption function
CsmAEADDecryptAlgorithmMode	Enumeration value that corresponds to the mode of the encryption algorithm of the cipher suite, e.-g. CRYPTO_ALGOMODE_GCM for TLS_PSK_WITH_AES_128_GCM_SHA256.
CsmAEADDecryptTagLength	Byte length of the key of the cipher suite's encryption function



Module Configuration Parameter	Value
CsmAEADDecryptAssociatedDataMaxLength	MAXIMUM-TRANSMISSION-UNIT of the ETHER-NET-COMMUNICATION-CONNECTOR associated with the local SOCKET-ADDRESS for which the TlsConnection was created.
CsmAEADDecryptCiphertextMaxLength	Same as CsmAEADDecryptAssociatedDataMaxLength
CsmAEADDecryptPlaintextMaxLength	Same as CsmAEADDecryptAssociatedDataMaxLength
CsmAEADDecryptKeyRef	Refers to the CsmKey container the CsmJob references via CsmJobKeyRef.

For SignatureGenerateCsmPrimitive containers a CsmSignatureGenerate sub container is created, the following parameters are configured:

Module Configuration Parameter	Value
CsmSignatureGenerateAlgorithmFamily	<p>The parameter is configured depending on the first curve that is associated with the cipher suite. Its value is set to one of the following values:</p> <ul style="list-style-type: none"> <li>▶ CRYPTO_ALGOFAM_ED25519 for the x25519 curve</li> <li>▶ CRYPTO_ALGOFAM_ED448 for the x448 curve</li> <li>▶ CRYPTO_ALGOFAM_ECCNIST for the NIST curves secp256r1, secp384r1, and secp521r1</li> </ul> <p><a href="#">Section 3.4.10.3, "Retrieval of curve information"</a> describes how the curves are retrieved from the cipher suite.</p>
CsmSignatureGenerateAlgorithmSecondaryFamily	<p>The parameter is configured depending on the first signature scheme that is associated with the cipher suite. Its value is set to one of the following values:</p> <ul style="list-style-type: none"> <li>▶ CRYPTO_ALGOFAM_SHA2_256 for the ecdsa_secp256r1_sha256 scheme</li> <li>▶ CRYPTO_ALGOFAM_SHA2_384 for the ecdsa_secp384r1_sha384 scheme</li> <li>▶ CRYPTO_ALGOFAM_SHA2_512 for the ecdsa_secp521r1_sha512 scheme</li> </ul>

Module Configuration Parameter	Value
	<a href="#">Section 3.4.10.4, “Retrieval of signature scheme information”</a> describes how the signature schemes are retrieved from the cipher suite.

### 3.4.10.3. Retrieval of curve information

Curves are retrieved by collecting all `CRYPTO-SERVICE-PRIMITIVE` elements that are referenced via `TLS-CRYPTO-CIPHER-SUITE/KEY-EXCHANGE-REFS`. A curve is identified by its `TLS/groupId` SDG which is compared to the IDs listed in [TLS Supported Groups](#).

### 3.4.10.4. Retrieval of signature scheme information

The signature schemes of a `TLS-CRYPTO-CIPHER-SUITE` are retrieved by collecting the comma separated IDs contained in the `TLS/signatureSchemes` SDG. These IDs are compared with the IDs listed in [TLS SignatureScheme](#).

## 3.4.11. Dcm

Configuration parameters	Mapping description
<code>DcmConfigSet</code>	Check if a <code>DcmConfigSet</code> container is created inside the module. If one already exists then an additional one is not created, otherwise a new <code>DcmConfigSet</code> container is created.
<code>DcmConfigSet/DcmDslProtocol/DcmDslProtocolRow</code>	<p>For each <code>DIAGNOSTIC-SERVICE-TABLE</code> that is assigned to the configured <code>ECU-INSTANCE</code>, one <code>DcmDslProtocolRow</code> container is created.</p> <p>A <code>DIAGNOSTIC-CONNECTION</code> refers to a PDU sent or received by the configured <code>ECU-INSTANCE</code> in the following cases:</p> <ul style="list-style-type: none"> <li>▶ The <code>DIAGNOSTIC-CONNECTION</code> references a <code>PDU-TRIGGERING</code> via <code>PERIODIC-RESPONSE-UUDT-REF</code> which in turn references an outgoing <code>I-PDU-PORT</code> of the <code>ECU-INSTANCE</code>. Moreover, the <code>PDU-TRIGGERING</code> must reference a PDU of type <code>DCM-I-PDU</code>.</li> <li>▶ The <code>DIAGNOSTIC-CONNECTION</code> references a <code>TP-CONNECTION-IDENT</code> element via <code>FUNCTIONAL-REQUEST-REF</code>, <code>RESPONSE-REF</code>, <code>RESPONSE-ON-EVENT-REF</code>, or <code>PHYSICAL-REQUEST-REF</code> which in turn refers to one or more PDUs that are sent or received by the configured <code>ECU-INSTANCE</code>.</li> </ul>

Configuration parameters	Mapping description
	<p>A TP-CONNECTION-IDENT refers to a PDU in one of the following cases:</p> <ul style="list-style-type: none"> <li>▶ The TP-CONNECTION-IDENT is aggregated by a CAN-TP-CONNECTION which refers to the PDU via TP-SDU-REF.</li> <li>▶ The TP-CONNECTION-IDENT is aggregated by a FLEXRAY-TP-CONNECTION which refers to the PDU either via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF.</li> <li>▶ The TP-CONNECTION-IDENT is aggregated by a LIN-TP-CONNECTION which refers to the PDU via LIN-TP-N-SDU-REF.</li> <li>▶ The TP-CONNECTION-IDENT is aggregated by a DO-IP-TP-CONNECTION, which refers to the PDU via TP-SDU-REF.</li> </ul>
DcmDslProtocol-Row/DcmDslProtocolID	<p>Each DiagnosticProtocol refers to at most one DiagnosticServiceTable and to a collection of DiagnosticConnection elements. The DiagnosticProtocolID parameter is configured to a certain category based on specific conditions for each category:</p> <ul style="list-style-type: none"> <li>▶ If the reference DiagnosticConnection.periodicResponseTp exists and the reference TpConnectionIdent.ident belongs to a CanTpConnection, or the DiagnosticConnection.periodicResponseUdt is not empty and the PhysicalChannel belongs to AbstractCanPhysicalChannel, then the DcmDslProtocolID is set to the DCM_PERIODIC_TRANS_ON_CAN category.</li> <li>▶ If the reference DiagnosticConnection.periodicResponseTp exists and the reference TpConnectionIdent.ident belongs to a FlexRayTpConnection, or the DiagnosticConnection.periodicResponseUdt is not empty and the PhysicalChannel belongs to FlexrayPhysicalChannel, then the DcmDslProtocolID is set to the DCM_PERIODIC_ON_FLEXRAY category.</li> <li>▶ If the reference DiagnosticConnection.periodicResponseTp exists and the reference TpConnectionIdent.ident belongs to a DoIpTpConnection, or the DiagnosticConnection.periodicResponseUdt is not empty and the PhysicalChannel belongs to EthernetPhysicalChannel, then the DcmDslProtocolID is set to the DCM_PERIODIC_ON_IP category.</li> <li>▶ If the reference DiagnosticConnection.responseOnEvent exists and the reference TpConnectionIdent.ident belongs to a CanTpConnection, then the DcmDslProtocolID is set to the DCM_ROE_ON_CAN category.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ If the reference <code>DiagnosticConnection.responseOnEvent</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>FlexRayTpConnection</code>, then the <code>DcmDslProtocolID</code> is set to <code>DCM_ROE_ON_FLEXRAY</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.responseOnEvent</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>DoIpTpConnection</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_ROE_ON_IP</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>CanTpConnection</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_UDS_ON_CAN</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>FlexRayTpConnection</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_UDS_ON_FLEXRAY</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>DoIpTpConnection</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_UDS_ON_IP</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>CanTpConnection</code> and if the <code>DiagnosticProtocol.protocolKind</code> is valid and contains <code>OBD</code> as a <code>String</code> or if the <code>DiagnosticServiceTable.serviceInstance</code> has one or more references to <code>DiagnosticServiceInstance</code> supported by <code>OBD</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_OBD_ON_CAN</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>FlexRayTpConnection</code> and if the <code>DiagnosticProtocol.protocolKind</code> is valid and contains <code>OBD</code> as a <code>String</code> or if the <code>DiagnosticServiceTable.serviceInstance</code> has one or more references to <code>DiagnosticServiceInstance</code> supported by <code>OBD</code>, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_OBD_ON_FLEXRAY</code> category.</li> <li>▶ If the reference <code>DiagnosticConnection.physicalRequest</code> exists and the reference <code>TpConnectionIdent.ident</code> belongs to a <code>DoIpTpConnection</code> and if the <code>DiagnosticProtocol.protocolKind</code> is valid and contains <code>OBD</code> as a <code>String</code> or if the <code>DiagnosticServiceTable.ser-</code></li> </ul>

Configuration parameters	Mapping description
	<code>viceInstance</code> has one or more references to <code>DiagnosticServiceInstance</code> supported by OBD, then the <code>DcmDslProtocolID</code> is set to the <code>DCM_OBD_ON_IP</code> category.
<code>DcmDslProtocolRow/DcmDslProtocolPriority</code>	<p>Each <code>DiagnosticProtocol</code> refers to at most one <code>DiagnosticServiceTable</code> and to a collection of <code>DiagnosticConnection</code> elements.</p> <p>If the reference <code>DiagnosticProtocol.serviceTable</code> exists, and the value of <code>DiagnosticProtocol.protocolKind</code> is identical to the value of <code>DiagnosticServiceTable.protocolKind</code>, then the <code>DcmDslProtocolPriority</code> is set and it represents the priority of the diagnostic protocol in comparison to other diagnostic protocols.</p>
<code>DcmDslBuffer</code>	<p>A <code>DcmDslBuffer</code> is configured.</p> <p>The size of <code>DcmDslBuffer</code> is set to have the default value, 8.</p>
<code>DcmDslProtocolRow/DcmDslProtocolSessionRef</code>	For every <code>DcmDslProtocolRow</code> a <code>DcmDslProtocolSessionRef</code> reference is set.
<code>DcmDslProtocolRow/DcmDemClientRef</code>	<p>The <code>DcmDemClientRef</code> reference in <code>DcmDslProtocolRow</code> is configured to refer to a <code>DemClient</code> container if at least one of the <code>DiagnosticServiceInstance</code> elements of the <code>DiagnosticServiceTable</code> is of any of the types listed below. The <code>DcmDemClientRef</code> is set to one of the following UDS supported services:</p> <ul style="list-style-type: none"> <li>▶ <code>DiagnosticReadDTCInformation</code> for <code>DIAGNOSTIC-READ-DTC-INFO</code>.</li> <li>▶ <code>DiagnosticClearDiagnosticInformation</code> for <code>DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFO</code>.</li> <li>▶ <code>DiagnosticControlDTCSetting</code> for <code>DIAGNOSTIC-CONTROL-DTC-SETTING</code>.</li> </ul> <p>The <code>DcmDemClientRef</code> is set to one of the following OBD supported services:</p> <ul style="list-style-type: none"> <li>▶ <code>DiagnosticRequestVehicleInfo</code> for <code>DIAGNOSTIC-REQUEST-VEHICLE-INFO</code>.</li> <li>▶ <code>DiagnosticRequestOnBoardMonitoringTestResults</code> for <code>DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS</code>.</li> <li>▶ <code>DiagnosticRequestPowertrainFreezeFrameData</code> for <code>DIAGNOSTIC-REQUEST-POWERTRAIN_FREEZE-FRAME-DATA</code>.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ DiagnosticRequestCurrentPowertrainDiagnosticData for DIAGNOSTIC-REQUEST-CURRENT-POWERTRAIN-DIAGNOSTIC-DATA.</li> <li>▶ DiagnosticRequestEmissionRelatedDTCPermanentStatus for DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC-PERMANENT-STATUS.</li> <li>▶ DiagnosticClearResetEmissionRelatedInfo for DIAGNOSTIC-CLEAR-RESET-EMISSION-RELATED-INFO.</li> </ul>
DcmDslProtocol-Row/DcmDslResponseOnEvent	<p>A DcmDslResponseOnEvent subcontainer is created if the DIAGNOSTIC-CONNECTION elements of the DIAGNOSTIC-SERVICE-TABLE refer to a PDU that the configured ECU-INSTANCE sends via RESPONSE-ON-EVENT-REF. The name of the container is set to &lt;PREFIX&gt;&lt;name&gt;_ResponseOnEvent, where &lt;name&gt; is the SHORT-NAME of the DIAGNOSTIC-CONNECTION.</p> <p>DcmDslRoeTxPduRef is configured to refer to the EcuC container that was created for the sent PDU.</p>
DcmDslProtocol-Row/DcmDslMainConnection	<p>A DcmDslMainConnection subcontainer is created for a DIAGNOSTIC-CONNECTION of a DIAGNOSTIC-SERVICE-TABLE if the DIAGNOSTIC-CONNECTION refers via FUNCTIONAL-REQUEST-REF, PHYSICAL-REQUEST-REF, or RESPONSE-REF to one or more TP-CONNECTION-IDENT elements that are in turn contained in TP-CONNECTION elements via which the configured ECU-INSTANCE sends or receives PDUs. The name of the container is set to &lt;PREFIX&gt;&lt;name&gt;_Main, where &lt;name&gt; is the SHORT-NAME of the DIAGNOSTIC-CONNECTION.</p> <p>DcmDslProtocolRxTesterSourceAddr is set to the diagnostic address of the remote node. The retrieval of that address depends on the type of TP-CONNECTION instance that contains the TP-CONNECTION-IDENT. Any type of TP-CONNECTION that is not listed below is not supported.</p> <ul style="list-style-type: none"> <li>▶ CAN-TP-CONNECTION instances: If CAN-TP-CONNECTION/TRANSMITTER does not refer to the configured ECU-INSTANCE, CAN-TP-CONNECTION/TRANSMITTER represents the remote node. If CAN-TP-CONNECTION/TRANSMITTER represents the configured ECU-INSTANCE, the remote node can only be determined if the CAN-TP-CONNECTION sends its PDUs to a physical address. In that case, the remote node is the first CAN-TP-NODE referenced by CAN-TP-CONNECTION/RECEIVER. The tester address is retrieved by taking the TP-ADDRESS value of the CAN-TP-ADDRESS entry which the remote node refers to via TP-ADDRESS.</li> </ul>

Configuration parameters	Mapping description
	<p>► <b>FLEXRAY-TP-CONNECTION instances:</b> If <b>FLEXRAY-TP-CONNECTION/TRANSMITTER</b> does not refer to the configured <b>ECU-INSTANCE</b>, <b>FLEXRAY-TP-CONNECTION/TRANSMITTER</b> represents the remote node. If <b>FLEXRAY-TP-CONNECTION/TRANSMITTER</b> represents the configured <b>ECU-INSTANCE</b>, the remote node can only be determined if the <b>FLEXRAY-TP-CONNECTION</b> sends its PDUs to a physical address. In that case, the remote node is the first <b>FLEXRAY-TP-NODE</b> referenced by <b>FLEXRAY-TP-CONNECTION/RECEIVER</b>. The tester address is retrieved by taking the <b>TP-ADDRESS/TP-ADDRESS</b> value which the remote <b>FLEXRAY-TP-NODE</b> refers to via <b>TP-ADDRESS</b>.</p> <p>► <b>DO-IP-TP-CONNECTION instances:</b> If the <b>TP-CONNECTION-IDENT</b> is referenced via <b>DIAGNOSTIC-CONNECTION/FUNCTIONAL-REQUEST</b> or <b>DIAGNOSTIC-CONNECTION/PHYSICAL-REQUEST</b>, <b>DO-IP-TP-CONNECTION/DO-IP-SOURCE-ADDRESS</b> represents the address of the tester node. If the <b>TP-CONNECTION-IDENT</b> is referenced via <b>DIAGNOSTIC-CONNECTION/RESPONSE</b>, <b>DO-IP-TP-CONNECTION/DO-IP-TARGET-ADDRESS</b> represents the address of the tester node. The actual tester address value is taken from <b>DO-IP-LOGIC-ADDRESS/ADDRESS</b>.</p> <p>All diagnostic addresses are collected for all <b>TP-CONNECTION-IDENT</b> elements which the <b>DIAGNOSTIC-CONNECTION</b> references via <b>FUNCTIONAL-REQUEST-REF</b>, <b>PHYSICAL-REQUEST-REF</b>, or <b>RESPONSE-REF</b>. The first element of the resulting list of diagnostic addresses is used for the configuration of <b>DcmDslProtocolRxTesterSourceAddr</b>.</p> <p>If the <b>DIAGNOSTIC-CONNECTION</b> references a PDU via <b>RESPONSE-ON-EVENT-REF</b>, <b>DcmDslROEConnectionRef</b> is configured to reference the <b>EcuC</b> container that was created for that PDU.</p>
DcmDslProtocol-Row/DcmDslMainConnection/DcmDslProtocolTx	<p>A <b>DcmDslProtocolTx</b> subcontainer is created if the <b>DIAGNOSTIC-CONNECTION</b> refers to one or more PDUs that the configured <b>ECU-INSTANCE</b> sends via <b>RESPONSE-REF</b>. The container name is <b>&lt;PREFIX&gt;&lt;name&gt;</b>, where <b>&lt;name&gt;</b> is the <b>SHORT-NAME</b> of the received PDU.</p> <p><b>DcmDslProtocolTxPduRef</b> is configured to reference the <b>EcuC</b> container that was created for the sent PDU.</p>
DcmDslProtocol-Row/DcmDslMainConnection/DcmDslProtocolRx	<p>A <b>DcmDslProtocolRx</b> subcontainer is created if the <b>DIAGNOSTIC-CONNECTION</b> refers to one or more PDUs that the configured <b>ECU-INSTANCE</b> receives via <b>FUNCTIONAL-REQUEST_REF</b> or <b>PHYSICAL_REQUEST-REF</b>. The container name is <b>&lt;PREFIX&gt;&lt;name&gt;</b>, where <b>&lt;name&gt;</b> is the <b>SHORT-NAME</b> name of the received PDU.</p>

Configuration parameters	Mapping description
	<p><b>DcmDslProtocolRxAddrType</b> is set to <code>DCM_PHYSICAL_TYPE</code> if the received PDU is referenced by the <code>DIAGNOSTIC-CONNECTION</code> via <code>PHYSICAL-REQUEST-REF</code>. It is set to <code>DCM_FUNCTIONAL_TYPE</code> if the received PDU is referenced by the <code>DIAGNOSTIC-CONNECTION</code> via <code>FUNCTIONAL-REQUEST-REF</code>.</p> <p><b>DcmDslProtocolRxPduRef</b> is configured to reference the <code>EcuC</code> container that was configured for the received PDU.</p> <p>If a <code>ComMChannel</code> container was created in the <code>ComM</code> for the <code>COMMUNICATION-CLUSTER</code> or the <code>ETHERNET-PHYSICAL-CHANNEL</code> via which the PDU is received, <b>DcmDslProtocolRxComMChannelRef</b> is configured to reference that <code>ComMChannel</code> container.</p>
DcmDslProtocol-Row/DcmDslPeriodicTransmission	<p>A <b>DcmDslPeriodicTransmission</b> subcontainer is created if the <code>DIAGNOSTIC-CONNECTION</code> elements of the <code>DIAGNOSTIC-SERVICE-TABLE</code> refer via <code>PERIODIC-RESPONSE-UUDT-REF</code> to one or more PDUs that the configured <code>ECU-INSTANCE</code> sends. The name of the container is set to <code>&lt;PREFIX&gt;&lt;name&gt;_PeriodicTransmission</code>, where <code>&lt;name&gt;</code> is the mangled name of the <code>DIAGNOSTIC-CONNECTION</code>.</p> <p>For each PDU that the configured <code>ECU-INSTANCE</code> sends in the context of the <code>DIAGNOSTIC-CONNECTION</code>, one <b>DcmDslPeriodicConnection</b> container is created. Its name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> name of the PDU. <b>DcmDslPeriodicTxPduRef</b> is configured to reference the <code>EcuC</code> container that was created for the sent PDU.</p>
DcmConfigSet/DcmDsp	<p><b>DcmDspMaxDidToRead</b> is set to the maximum <code>MAX-DID-TO-READ</code> value of any <code>DIAGNOSTIC-READ-DATA-BY-IDENTIFIER-CLASS</code> that is referenced by a <code>DIAGNOSTIC-READ-DATA-BY-IDENTIFIER</code> element of the imported <code>ECU-INSTANCE</code>.</p> <p><b>DcmDspMaxPeriodicDidToRead</b> is set to the maximum <code>MAX-PERIODIC-DID-TO-READ</code> value of any <code>DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID-CLASS</code> that is referenced by a <code>DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID</code> element of the imported <code>ECU-INSTANCE</code>.</p> <p>For the configuration of <b>DcmDspDDDDIDcheckPerSourceDID</b>, the <code>CHECK-PER-SOURCE-ID</code> values of all <code>DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER-CLASS</code> entities referenced by <code>DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER</code> elements of the imported <code>ECU-INSTANCE</code> are collected. If at least one <code>CHECK-PER-SOURCE-ID</code> value is set to <code>true</code>, <b>DcmDspDDDDIDcheckPerSourceDID</b> is set to <code>true</code>. If all <code>CHECK-PER-SOURCE-ID</code></p>



Configuration parameters	Mapping description
	values are set to false, then DcmDspDDDIDcheckPerSourceDID is set to false.
DcmConfigSet/DcmDsp/DcmDspMemory	A single DcmDspMemory sub container is created for all the DIAGNOSTIC-READ-MEMORY-BY-ADDRESS and all the DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS services referenced by a DIAGNOSTIC-SERVICE-TABLE for the current ECU.
DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo	For each different DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS or a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service, belonging to a DIAGNOSTIC-SERVICE-TABLE, a single DcmDspMemoryIdInfo is created.  DcmDspMemoryIdValue is set to the ID value of the DIAGNOSTIC-MEMORY-IDENTIFIER.
DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeInfo	A single DcmDspWriteMemoryRangeInfo subcontainer is created for each DIAGNOSTIC-MEMORY-IDENTIFIER that is referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS.  DcmDspWriteMemoryRangeHigh is set to the MEMORY-HIGH-ADDRESS value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.  DcmDspWriteMemoryRangeLow is set to the MEMORY-LOW-ADDRESS value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.  DcmDspWriteMemoryRangeSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.
DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeInfo	A single DcmDspReadMemoryRangeInfo subcontainer is created for each DIAGNOSTIC-MEMORY-IDENTIFIER that is referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS.  DcmDspReadMemoryRangeHigh is set to the MEMORY-HIGH-ADDRESS value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.  DcmDspReadMemoryRangeLow is set to the MEMORY-LOW-ADDRESS value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.

Configuration parameters	Mapping description
	DcmDspReadMemoryRangeSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.
DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeByLabelInfo	<p>A single DcmDspWriteMemoryRangeByLabelInfo subcontainer is created for each DIAGNOSTIC-MEMORY-IDENTIFIER that is referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS.</p> <p>DcmDspWriteMemoryRangeByLabelHigh is set to the MEMORY-HIGH-ADDRESS-LABEL value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.</p> <p>DcmDspWriteMemoryRangeByLabelLow is set to the MEMORY-LOW-ADDRESS-LABEL value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.</p> <p>DcmDspWriteMemoryRangeSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS service.</p>
DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeByLabelInfo	<p>A single DcmDspReadMemoryRangeByLabelInfo subcontainer is created for each DIAGNOSTIC-MEMORY-IDENTIFIER that is referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS.</p> <p>DcmDspReadMemoryRangeByLabelHigh is set to the MEMORY-HIGH-ADDRESS-LABEL value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.</p> <p>DcmDspReadMemoryRangeByLabelLow is set to the MEMORY-LOW-ADDRESS-LABEL value of the DIAGNOSTIC-MEMORY-IDENTIFIER referenced by a DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.</p> <p>DcmDspReadMemoryRangeSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-READ-MEMORY-BY-ADDRESS service.</p>
DcmConfigSet/DcmDsp/DcmDspComCon-	A DcmDspComControlAllChannel container is created for every different DIAGNOSTIC-ALL-CHANNEL element of a DIAGNOSTIC-COM-CON-

Configuration parameters	Mapping description
trol/DcmDspComControlAllChannel	<p>TROL-CLASS of the current ECU. The container name is &lt;name&gt;&lt;SUFFIX&gt;, where &lt;name&gt; is derived from the short name of the DIAGNOSTIC-ALL-CHANNEL element and the &lt;SUFFIX&gt; is _AC_ and the number of the current container created.</p> <p>DcmDspAllComMChannelRef is set to DIAGNOSTIC-COM-CONTROL-CLASS/ALL-CHANNEL reference.</p>
DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSpecificChannel	<p>A DcmDspComControlSpecificChannel container is created for every different pair of SPECIFIC-CHANNEL and SUBNET-NUMBER elements from a DIAGNOSTIC-COM-CONTROL-SPECIFIC-CHANNEL of a DIAGNOSTIC-COM-CONTROL-CLASS of the current ECU. The container name is &lt;name&gt;&lt;SUFFIX&gt;, where &lt;name&gt; is derived from the short name of the SPECIFIC-CHANNEL element and the &lt;SUFFIX&gt; is _SC_ and the number of the current container created.</p> <p>DcmDspSubnetNumber is set to the value of DIAGNOSTIC-COM-CONTROL-SPECIFIC-CHANNEL/SUBNET-NUMBER.</p> <p>DcmDspSpecificComMChannelRef is set to DIAGNOSTIC-COM-CONTROL-CLASS/SPECIFIC-CHANNEL reference.</p>
DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSubNode	<p>A DcmDspComControlSubNode container is created for every different pair of SUBNODE-CHANNEL and SUBNODE-NUMBER elements from a DIAGNOSTIC-COM-CONTROL-SUBNODE-CHANNEL of a DIAGNOSTIC-COM-CONTROL-CLASS of the current ECU. The container name is &lt;name&gt;&lt;SUFFIX&gt;, where &lt;name&gt; is derived from the short name of the SUBNODE-CHANNEL element and the &lt;SUFFIX&gt; is _SubNode_ and the number of the current container created.</p> <p>DcmDspComControlSubNodeId is set to the value of DIAGNOSTIC-COM-CONTROL-SUBNODE-CHANNEL/SUBNODE-NUMBER.</p> <p>DcmDspComControlSubNodeComMChannelRef is set to DIAGNOSTIC-COM-CONTROL-SUBNODE-CHANNEL/SUBNODE-CHANNEL reference.</p>
DcmConfigSet/DcmDsp/DcmDspSession	<p>A DcmDspSession subcontainer is created for every DIAGNOSTIC-SESSION referenced by a DIAGNOSTIC-ACCESS-PERMISSION associated with any DiagnosticServiceInstance of the current ECU. For more information on how the DIAGNOSTIC-ACCESS-PERMISSION of a DiagnosticServiceInstance is retrieved, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>. The container name is DIAGNOSTIC-SESSION/SHORT-NAME of the DIAGNOSTIC-SESSION-CONTROL.</p>

Configuration parameters	Mapping description
	<p>DcmDspSessionForBoot is set to DIAGNOSTIC-SESSION/JUMP-TO-BOOT-LOADER.</p> <p>DcmDspSessionLevel is set to DIAGNOSTIC-SESSION/ID.</p> <p>DcmDspSessionP2ServerMax is set to DIAGNOSTIC-SESSION/P-2-SERVER-MAX.</p> <p>DcmDspSessionP2StarServerMax is set to DIAGNOSTIC-SESSION/P-2-STAR-SERVER-MAX.</p>
DcmConfigSet/DcmDsp/DcmDspSecurity	<p>A DcmDspSecurity subcontainer is created for every DIAGNOSTIC-SECURITY-ACCESS referenced by a DIAGNOSTIC-ACCESS-PERMISSION associated with any DiagnosticServiceInstance of the current ECU. For more information on how the DIAGNOSTIC-ACCESS-PERMISSION of a DiagnosticServiceInstance is retrieved, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>. The container name is DIAGNOSTIC-SECURITY-LEVEL/SHORT-NAME of the DIAGNOSTIC-SECURITY-ACCESS.</p> <p>DcmDspSecurityADRSIZE is set to DIAGNOSTIC-SECURITY-LEVEL/ACCESS-DATA-RECORD-SIZE.</p> <p>DcmDspSecurityDelayTimeOnBoot is set to DIAGNOSTIC-SECURITY-LEVEL/SECURITY-DELAY-TIME.</p> <p>DcmDspSecurityKeySize is set to DIAGNOSTIC-SECURITY-LEVEL/KEY-SIZE.</p> <p>DcmDspSecurityLevel is set to (DIAGNOSTIC-SECURITY-ACCESS/REQUEST-SEED-ID + 1) / 2.</p> <p>DcmDspSecurityNumAttDelay is set to DIAGNOSTIC-SECURITY-LEVEL/NUM-FAILED-SECURITY-ACCESS.</p> <p>DcmDspSecuritySeedSize is set to DIAGNOSTIC-SECURITY-LEVEL/SEED-SIZE.</p>
DcmConfigSet/DcmDsp/DcmDspDataUsePort	<p>A DcmDspDataUsePort subcontainer indicates the interface that shall be used to access the data.</p> <p>USE_DATA_ASYNC_CLIENT_SERVER if DiagnosticProcessingStyleEnum is equal to processingStyleAsynchronous. The DiagnosticProcessingStyleEnum is taken from the DiagnosticServiceSwMapping that has a SwcServiceDependency taken directly from</p>

Configuration parameters	Mapping description
	<p>MappedFlatSwcServiceDependency or indirectly from MappedSwcServiceDependency with a target value taken from the SwcServiceDependencyInCompositionRef.</p> <p>USE_DATA_ASYNC_FNC if DiagnosticServiceSwMapping has a BswServiceDependency and DiagnosticProcessingStyleEnum is equal to processingStyleAsynchronous.</p> <p>USE_DATA_SYNC_CLIENT_SERVER if DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous. The DiagnosticProcessingStyleEnum is taken from the DiagnosticServiceSwMapping that has a SwcServiceDependency taken directly from MappedFlatSwcServiceDependency or indirectly from MappedSwcServiceDependency with a target value taken from the SwcServiceDependencyInCompositionRef.</p> <p>USE_DATA_SYNC_FNC if DiagnosticServiceSwMapping has a BswServiceDependency and DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous.</p> <p>If USE_DATA_ASYNC_CLIENT_SERVER or USE_DATA_SYNC_CLIENT_SERVER is set on DcmDspDataUsePort then the value of the parameter /Dcm/DcmConfigSet/DcmGeneral/DcmRteUsage is set to true.</p>
DcmConfigSet/DcmDsp/DcmDspControlDTCSetting	<p>A DcmDspControlDTCSetting subcontainer is created for the first occurrence of DIAGNOSTIC-CONTROL-DTC-SETTING. The container name is set to DcmDspControlDTCSetting.</p> <p>DcmSupportDTCSettingControlOptionRecord is set to DIAGNOSTIC-CONTROL-DTC-SETTING-CLASS/CONTROL-OPTION-RECORD-PRESENT referenced by DIAGNOSTIC-CONTROL-DTC-SETTING.</p>
DcmConfigSet/DcmDsp/DiagnosticReadDataByPeriodicID	<p>A DiagnosticReadDataByPeriodicID subcontainer is created for the first occurrence of DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID. The container name is set to DiagnosticReadDataByPeriodicID.</p> <p>DiagnosticPeriodicRate is set to DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID-CLASS/DIAGNOSTIC-PERIODIC-RATE referenced by DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID. referenced by DIAGNOSTIC-PERIODIC-TRANSMISSION</p> <p>DcmDspMaxPeriodicDidScheduler is set to maximum number of periodicDataIdentifiers that can be scheduled concurrently DIAGNOSTIC-READ-</p>

Configuration parameters	Mapping description
	DATA-BY-PERIODIC-ID-CLASS/SCHEDULER-MAX-NUMBER referenced by DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID.
DcmConfigSet/DcmDsp/DcmDspPeriodicTransmission	For the configuration of DcmDspPeriodicTransmissionSlowRate, DcmDspPeriodicTransmissionMediumRate, and DcmDspPeriodicTransmissionFastRate, all DIAGNOSTIC-PERIODIC-RATE elements of all DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID-CLASS elements referenced by DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID of the imported ECU-INSTANCE are collected and grouped according to their PERIODIC-RATE-CATEGORY value. DcmDspPeriodicTransmissionSlowRate is set to the smallest value of category PERIODIC-RATE-SLOW. DcmDspPeriodicTransmissionMediumRate is set to the smallest value of category PERIODIC-RATE-MEDIUM. DcmDspPeriodicTransmissionFastRate is set to the smallest value of category PERIODIC-RATE-FAST.
DcmConfigSet/DcmDsp/DiagnosticDynamicallyDefineDataIdentifier	<p>A DiagnosticDynamicallyDefineDataIdentifier subcontainer is created for the first occurrence of DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER. The container name is set to DiagnosticDynamicallyDefineDataIdentifier.</p> <p>DcmDDDIDStorage is set to true if all DiagnosticDynamicallyDefineDataIdentifier are stored NON-VOLATILE. DcmDDDIDStorage is set to false if any of DiagnosticDynamicallyDefineDataIdentifier are stored as VOLATILE. DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER-CLASS/CONFIGURATION-HANDLING referenced by DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER.</p>
DcmConfigSet/DcmDsp/DiagnosticRequestControlOfOnBoardDevice	<p>A DiagnosticRequestControlOfOnBoardDevice subcontainer is created for the first occurrence of DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE. The container name is set to DiagnosticRequestControlOfOnBoardDevice.</p> <p>DiagnosticTestRoutineIdentifier is set to DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE/DIAGNOSTIC-TEST-ROUTINE-IDENTIFIER.</p> <p>For every DiagnosticTestRoutineIdentifier DcmDspRequestControlTestId is set to DIAGNOSTIC-TEST-ROUTINE-IDENTIFIER/ID. DcmDspRequestControlInBufferSize is set to DIAGNOSTIC-TEST-ROUTINE-IDENTIFIER/REQUEST-DATA-SIZE. DcmDspRequestControlOutBufferSize is set to DIAGNOSTIC-TEST-ROUTINE-IDENTIFIER/RESPONSE-DATA-SIZE.</p>

Configuration parameters	Mapping description
DcmConfigSet/DcmDsp/DcmDspData	<p><b>For every</b> DIAGNOSTIC-DATA-IDENTIFIER/DIAGNOSTIC-PARAMETER/DIAGNOSTIC-DATA-ELEMENT <b>referenced by</b> DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER and DIAGNOSTIC-READ-DATA-BY-IDENTIFIER <b>service</b> instance which belongs to the imported ECU-INSTANCE, a DcmDspData container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;PREFIX&gt; is DspData_ + I where I is a counter and &lt;name&gt; is the DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME.</p> <p>DcmDspDataType is set depending on SW-BASE-TYPE/BASE-TYPE-ENCODING and SW-BASE-TYPE/BASE-TYPE-SIZE:</p> <ul style="list-style-type: none"> <li>▶ <b>UINT8</b> for BASE-TYPE-ENCODING NONE and BASE-TYPE-SIZE 8</li> <li>▶ <b>UINT16</b> for BASE-TYPE-ENCODING NONE and BASE-TYPE-SIZE 16</li> <li>▶ <b>UINT32</b> for BASE-TYPE-ENCODING NONE and BASE-TYPE-SIZE 32</li> <li>▶ <b>SINT8</b> for BASE-TYPE-ENCODING 2C and BASE-TYPE-SIZE 8</li> <li>▶ <b>SINT16</b> for BASE-TYPE-ENCODING 2C and BASE-TYPE-SIZE 16</li> <li>▶ <b>SINT32</b> for BASE-TYPE-ENCODING 2C and BASE-TYPE-SIZE 32</li> <li>▶ <b>BOOLEAN</b> for BASE-TYPE-ENCODING BOOLEAN and BASE-TYPE-SIZE 1 or BASE-TYPE-SIZE 8</li> </ul> <p>DcmDspDataSize is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * BASE-TYPE-SIZE. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspDataSize value is set to DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS*8. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspDataSize is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16. DcmDspDataInfoRef is set to DcmConfigSet/DcmDsp/DcmDspDataInfo/DcmDspDataInfo_0 if the ArraySizeSemantics value of the DiagnosticDataElement is FIXED-SIZE else it references DcmConfigSet/DcmDsp/DcmDspDataInfo/DcmDspDataInfo_1 if the ArraySizeSemantics value of the DiagnosticDataElement is VARIABLE-SIZE.</p> <p>DcmDspDataEndianness is set to BIG_ENDIAN if BASE-TYPE/BYTE-ORDER is defined as MOST-SIGNIFICANT-BYTE-FIRST or set to LITTLE_ENDIAN if</p>

Configuration parameters	Mapping description
	BASE-TYPE/BYTE-ORDER is defined as MOST-SIGNIFICANT-BYTE-LAST or set to OPAQUE if BASE-TYPE/BYTE-ORDER is defined as OPAQUE.
DcmConfigSet/DcmDsp/DcmDspDidInfo	<p>For each different Diagnostic-Data-Identifier referenced by a DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER, DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER or DIAGNOSTIC-READ-DATA-BY-IDENTIFIER service instance which belongs to the imported ECU-INSTANCE a single DcmDspDidInfo is created if DcmDspDidDynamicallyDefined, DcmDspDDDIDMaxElements, and DcmDspDidAccess parameters are the same in every service instance. If the service instance is a DIAGNOSTIC-IO-CONTROL then the following parameters are also checked: FreezeCurrentState, ResetToDefault, and ShortTermAdjustment. If they are different, a new DcmDspDidInfo is created. If any two of the above service instances refers a different Diagnostic-Data-Identifier, however have the same AccessPermission then an extra check is done in order to see if there is already a matching DcmDspDidInfo that can be used or if a new one needs to be created. The information contained by DcmDspDidInfo will not be the same. The container name is set to DcmDspDidInfo_&lt;suffix&gt;, where the &lt;suffix&gt; is set to the number of the container created.</p> <p>DcmDspDidDynamicallyDefined is set to true or false according to DataIdentifier from DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER.</p> <p>DcmDspDDDIDMaxElements is set to the maximum MAX-SOURCE-ELEMENT value of all DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER elements that refer via DYNAMICALLY-DEFINE-DATA-IDENTIFIER-CLASS-REF to the DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER-CLASS element for which the DcmDspDidInfo container has been created.</p>
DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidAccess/DcmDspDidRead	<p>If at least one DIAGNOSTIC-READ-DATA-BY-IDENTIFIER exists, a DcmDspDidRead container is created.</p> <p>DcmDspDidReadSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-READ-DATA-BY-IDENTIFIER references. For more information, see also <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p> <p>DcmDspDidReadSessionRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-READ-DATA-BY-IDENTIFIER references. For more information, see also <a href="#">Sec-</a></p>



Configuration parameters	Mapping description
	<a href="#">tion 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a> .
DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidAccess/DcmDspDidWrite	<p>If at least one DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER exists, a DcmDspDidWrite container is created.</p> <p>DcmDspDidWriteSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p>DcmDspDidWriteSessionRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidAccess/DcmDspDidControl	<p>If at least one DIAGNOSTIC-IO-CONTROL exists, a DcmDspDidControl container is created.</p> <p>DcmDspDidControlSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-IO-CONTROL references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p>DcmDspDidControlSessionRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-IO-CONTROL references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspDid	For every DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER and DIAGNOSTIC-READ-DATA-BY-IDENTIFIER service instance which belongs to the imported ECU-INSTANCE, a DcmDspDid container is created. The container name is <PREFIX><name>, where <PREFIX> is DspDid_ + I where I is a counter and <name> is set to DIAGNOSTIC-DATA-IDENTIFIER/SHORT-NAME

Configuration parameters	Mapping description
	<p>of DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER and DIAGNOSTIC-READ-DATA-BY-IDENTIFIER.</p> <p>DcmDspDidIdentifier is set to DIAGNOSTIC-DATA-IDENTIFIER/ID.</p> <p>DcmDspDidInfoRef references the corresponding DcmDspDidInfo container.</p>
DcmConfigSet/DcmDsp/DcmDspDid/DcmDspDidSignal	<p>For every DIAGNOSTIC-DATA-IDENTIFIER/DIAGNOSTIC-PARAMETER referenced by DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER and DIAGNOSTIC-READ-DATA-BY-IDENTIFIER service instance which belongs to the imported ECU-INSTANCE, a DcmDspDidSignal container is created. The container name is DcmDspDidSignal_&lt;suffix&gt;, where &lt;suffix&gt; is set to the zero-based position index of the DIAGNOSTIC-PARAMETER within the DIAGNOSTIC-DATA-IDENTIFIER.</p> <p>DcmDspDidDataPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspDidDataRef references the DcmDspDid created which was created to represent the DIAGNOSTIC-DATA-ELEMENT of the current DIAGNOSTIC-PARAMETER.</p>
DcmConfigSet/DcmDsp/DcmDspRoutine	<p>For every DIAGNOSTIC-ROUTINE-CONTROL service instance which belongs to the imported ECU-INSTANCE, a DcmDspRoutine container is created. The container name is set to DIAGNOSTIC-ROUTINE/SHORT-NAME of DIAGNOSTIC-ROUTINE-CONTROL.</p> <p>DcmDspRoutineIdentifier is set to DIAGNOSTIC-ROUTINE/ID.</p> <p>DcmDspRoutineInfoRef references the corresponding DcmDspRoutineInfo container.</p> <p>DcmDspRoutineUsePort is set to true if the DIAGNOSTIC-ROUTINE is configured for a SWC-SERVICE-DEPENDENCY based on a MAPPED-FLAT-SWC-SERVICE-DEPENDENCY or a MAPPED-SWC-SERVICE-DEPENDENCY via DIAGNOSTIC-SERVICE-SW-MAPPING. If at least one DcmDspRoutineUsePort is set to true, the value of the parameter /Dcm/DcmConfigSet/DcmGeneral/DcmRteUsage is set to true.</p> <p>Is set to true, if the DiagnosticRoutine has a DiagnosticRequestRoutineResults configured.</p> <p>Is set to true, if the DiagnosticRoutine has a DiagnosticStopRoutine configured.</p>

Configuration parameters	Mapping description
	DcmDspRoutineFixedLength is set to false if all the signals referenced by the DIAGNOSTIC-ROUTINE are of type VARIABLE_SIZE. Otherwise, the DcmDspRoutineFixedLength is set to true.
DcmConfigSet/DcmDsp/DcmDspRoutineInfo	For every DIAGNOSTIC-ROUTINE-CONTROL service instance which belongs to the imported ECU-INSTANCE, a DcmDspRoutineInfo container is created. The container name is set to DcmDspRoutineInfo_<suffix>, where <suffix> is set to DIAGNOSTIC-ROUTINE/ID of DIAGNOSTIC-ROUTINE-CONTROL.
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspRoutineAuthorization	<p>If at least one DIAGNOSTIC-ROUTINE-CONTROL exists, a DcmDspRoutineAuthorization container is created.</p> <p>DcmDspRoutineSecurityLevelRef references all the corresponding DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-ROUTINE-CONTROL references. For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p> <p>DcmDspRoutineSessionRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-ROUTINE-CONTROL references. For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal	<p>For every request DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/START a DcmDspStartRoutineInSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p> <p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * BASE-TYPE-SIZE. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS*8. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE.</p>

Configuration parameters	Mapping description
	<p>For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal	<p>For every response DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/START a DcmDspStartRoutineOutSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p> <p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * BASE-TYPE-SIZE. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS*8. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspRoutineStopIn/DcmDspRoutineStopInSignal	<p>For every request DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/STOP a DcmDspRoutineStopInSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p> <p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * BASE-TYPE-SIZE. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT</p>

Configuration parameters	Mapping description
	<p>is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS*8. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspRoutineStopOut/DcmDspRoutineStopOutSignal	<p>For every response DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/STOP a DcmDspRoutineStopOutSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p> <p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * BASE-TYPE-SIZE. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS*8. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspRoutineRequestResIn/DcmDspRoutineRequestResInSignal	<p>If the REQUESTS tag exists inside the DiagnosticExtract, then for every request DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/REQUEST-RESULT a DcmDspRoutineRequestResInSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p>

Configuration parameters	Mapping description
tineRequestResInSignal	<p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value <math>\text{DIAGNOSTIC-DATA-ELEMENT}/\text{MAX-NUMBER-OF-ELEMENTS} * \text{BASE-TYPE-SIZE}</math>. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to <math>\text{DIAGNOSTIC-DATA-ELEMENT}/\text{MAX-NUMBER-OF-ELEMENTS} * 8</math>. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p> <p>DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p>
DcmConfigSet/DcmDsp/DcmDspRoutineInfo/DcmDspRoutineRequestResOut/DcmDspRoutineRequestResOutSignal	<p>For every response DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-ROUTINE/REQUEST-RESULT a DcmDspRoutineRequestResOutSignal container is created. The container name is set to DIAGNOSTIC-DATA-ELEMENT/SHORT-NAME aggregated by DIAGNOSTIC-PARAMETER</p> <p>DcmDspRoutineSignalLength is set depending on the ARRAY-SIZE-SEMANTICS if it is FIXED-SIZE. Then the BASE-TYPE-SIZE associated with the DIAGNOSTIC-DATA-ELEMENT is obtained and set to the value <math>\text{DIAGNOSTIC-DATA-ELEMENT}/\text{MAX-NUMBER-OF-ELEMENTS} * \text{BASE-TYPE-SIZE}</math>. If the ARRAY-SIZE-SEMANTICS value of the DIAGNOSTIC-DATA-ELEMENT is VARIABLE-SIZE, the DcmDspRoutineSignalLength value is set to <math>\text{DIAGNOSTIC-DATA-ELEMENT}/\text{MAX-NUMBER-OF-ELEMENTS} * 8</math>. If DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS is missing then its value is considered 0. If ARRAY-SIZE-SEMANTICS is not set then the value of DcmDspRoutineSignalLength is set to the size given by BASE-TYPE-SIZE. For example 8 for UINT8, 16 for UINT16 etc. See also <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a>.</p> <p>DcmDspRoutineSignalPos is set to DIAGNOSTIC-PARAMETER/BIT-OFFSET.</p>

Configuration parameters	Mapping description
	DcmDspRoutineSignalType is set based on <a href="#">Section 3.4.11.23, “Determining the type for a routine signal”</a> .
DcmConfigSet/DcmDsp/DcmDspVehInfo	<p>For every DIAGNOSTIC-REQUEST-VEHICLE-INFO service instance which belongs to the imported ECU-INSTANCE, a DcmDspVehInfo container is created. The container name is set to DcmDspVehInfo&lt;suffix&gt;, where &lt;suffix&gt; is an underscore followed by VEHICLE-INFO-TYPE/ID.</p> <p>DcmDspVehInfoInfoType is set with the value of VEHICLE-INFO-TYPE/ID.</p>
DcmConfigSet/DcmDsp/DcmDspVehInfo/DcmDspVehInfoData	<p>The DcmDspVehInfoData subcontainer is created for every request DIAGNOSTIC-PARAMETER aggregated by DIAGNOSTIC-REQUEST-VEHICLE-INFO. The name of DcmDspVehInfoData sub container is derived from the short name of the DIAGNOSTIC-DATA-ELEMENT aggregated by the DIAGNOSTIC-PARAMETER.</p> <p>DcmDspVehInfoDataOrder parameter will be take the value from DATA-ELEMENT/bitOffset that it is bellonged to DIAGNOSTIC-INFO-TYPE aggregated by the DIAGNOSTIC-REQUEST-VEHICLE-INFO .</p> <p>DcmDspVehInfoDataReadFnc parameter will take the value based on the DIAGNOSTIC-SERVICE-SW-MAPPING that has set the reference to a MAPPED-BSW-SERVICE-DEPENDENCY . The BSW-SERVICE-DEPENDENCY should have a ROLE-BASED-BSW-MODULE-ENTRY-ASSIGNMENT that in turn has attribute role set to Xxx_GetInfotypeValueData and points to a BSW-MODULE-ENTRY .</p> <p>DcmDspVehInfoDataSize parameter will take the value from DATA-ELEMENT/numberOfElements that it is bellonged to DIAGNOSTIC-INFO-TYPE aggregated by the DIAGNOSTIC-REQUEST-VEHICLE-INFO .</p> <p>DcmDspVehInfoDataUsePort parameter will be set to true if the reference DIAGNOSTIC-SERVICE-SW-MAPPING/mappedSwServiceDepenedency exists or will be set to false if the reference DIAGNOSTIC-SERVICE-SW-MAPPING/mappedBswServiceDependency exists. The DIAGNOSTIC-SERVICE-SW-MAPPING shall have a link with a DIAGNOSTIC-DATA-ELEMENT that is aggregated by a DIAGNOSTIC-REQUEST-VEHICLE-INFO .</p>
DcmConfigSet/DcmDsd/DcmDsdService	For every <i>first</i> occurrence of a DIAGNOSTIC-READ-DATA-BY-IDENTIFIER, DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER, DIAGNOSTIC-SESSION-CONTROL, DIAGNOSTIC-SECURITY-ACCESS, DIAGNOSTIC-ROUTINE-CONTROL, DIAGNOSTIC-READ-DTC-INFORMATION, DIAGNOSTIC-ECU-RESET,

Configuration parameters	Mapping description
eTable/DcmDsdService	<p>DIAGNOSTIC-COM-CONTROL, DIAGNOSTIC-DIAGNOSTIC-REQUEST-DOWNLOAD, DIAGNOSTIC-REQUEST-UPLOAD, DIAGNOSTIC-DATA-TRANSFER, DIAGNOSTIC-REQUEST-TRANSFER-EXIT, DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID, DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER, DIAGNOSTIC-IO-CONTROL, DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS, DIAGNOSTIC-READ-MEMORY-BY-ADDRESS, DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFORMATION, DIAGNOSTIC-REQUEST-POWERTRAIN_FREEZE-FRAME-DATA, DIAGNOSTIC-REQUEST-CURRENT-POWERTRAIN-DIAGNOSTIC-DATA, DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC-PERMANENT-STATUS, DIAGNOSTIC-CONTROL-DTC-SETTING, DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS, DIAGNOSTIC-REQUEST-VEHICLE-INFO, DIAGNOSTIC-CLEAR-RESET-EMISSION-RELATED-INFO <b>service instance which belongs to the imported ECU-INSTANCE, a DcmDsdService container is created. The container name is set to one of the following:</b></p> <ul style="list-style-type: none"> <li>▶ DiagnosticReadDataByIdentifier <b>for</b> DIAGNOSTIC-READ-DATA-BY-IDENTIFIER.</li> <li>▶ DiagnosticWriteDataByIdentifier <b>for</b> DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER.</li> <li>▶ DiagnosticSessionControl <b>for</b> DIAGNOSTIC-SESSION-CONTROL.</li> <li>▶ DiagnosticSecurityAccess <b>for</b> DIAGNOSTIC-SECURITY-ACCESS.</li> <li>▶ DiagnosticRoutineControl <b>for</b> DIAGNOSTIC-ROUTINE-CONTROL.</li> <li>▶ DiagnosticReadDTCInformation <b>for</b> DIAGNOSTIC-READ-DTC-INFORMATION.</li> <li>▶ DiagnosticEcuReset <b>for</b> DIAGNOSTIC-ECU-RESET.</li> <li>▶ DiagnosticComControl <b>for</b> DIAGNOSTIC-COM-CONTROL.</li> <li>▶ DiagnosticRequestVehicleInfo <b>for</b> DIAGNOSTIC-REQUEST-VEHICLE-INFO.</li> <li>▶ DiagnosticRequestOnBoardMonitoringTestResults <b>for</b> DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS.</li> <li>▶ DiagnosticRequestDownload <b>for</b> DIAGNOSTIC-REQUEST-DOWNLOAD.</li> <li>▶ DiagnosticRequestUpload <b>for</b> DIAGNOSTIC-REQUEST-UPLOAD.</li> <li>▶ DiagnosticDataTransfer <b>for</b> DIAGNOSTIC-DATA-TRANSFER.</li> <li>▶ DiagnosticRequestTransferExit <b>for</b> DIAGNOSTIC-REQUEST-TRANSFER-EXIT.</li> </ul>



Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ DiagnosticReadDataByPeriodicIdentifier <b>for</b> DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID.</li> <li>▶ DiagnosticWriteMemoryByAddress <b>for</b> DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS.</li> <li>▶ DiagnosticReadMemoryByAddress <b>for</b> DIAGNOSTIC-READ-MEMORY-BY-ADDRESS.</li> <li>▶ DiagnosticIOControl <b>for</b> DIAGNOSTIC-IO-CONTROL.</li> <li>▶ DiagnosticControlDTCSetting <b>for</b> DIAGNOSTIC-CONTROL-DTC-SETTING.</li> <li>▶ DiagnosticDynamicallyDefineDataIdentifier <b>for</b> DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER.</li> <li>▶ DiagnosticRequestControlOfOnBoardDevice <b>for</b> DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE.</li> <li>▶ DiagnosticClearDiagnosticInformation <b>for</b> DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFORMATION.</li> <li>▶ DiagnosticRequestPowertrainFreezeFrameData <b>for</b> DIAGNOSTIC-REQUEST-POWERTRAIN_FREEZE-FRAME-DATA.</li> <li>▶ DiagnosticRequestCurrentPowertrainDiagnosticData <b>for</b> DIAGNOSTIC-REQUEST-CURRENT-POWERTRAIN-DIAGNOSTIC-DATA.</li> <li>▶ DiagnosticRequestEmissionRelatedDTCPermanentStatus <b>for</b> DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC-PERMANENT-STATUS.</li> <li>▶ DiagnosticClearResetEmissionRelatedInfo <b>for</b> DIAGNOSTIC-CLEAR-RESET-EMISSION-RELATED-INFO.</li> </ul> <p><b>For every <i>first</i> occurrence of a DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC service instance which belongs to the imported ECU-INSTANCE, two DcmDsdService containers are created. The containers name are set to the following:</b></p> <ul style="list-style-type: none"> <li>▶ DiagnosticRequestEmissionRelatedDTC_03 <b>for</b> DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC.</li> <li>▶ DiagnosticRequestEmissionRelatedDTC_07 <b>for</b> DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC.</li> </ul> <p><b>From the second occurrence onwards, only the configuration parameters DcmDsdSidTabSessionLevelRef and DcmDsdSidTabSecurityLevel-</b></p>

Configuration parameters	Mapping description
	<p>Ref are updated. If AccessPermissionValidity is set to ACCESS-PERMISSION-SERVICE-INSTANCE then DcmDsdSidTabSessionLevelRef and DcmDsdSidTabSecurityLevelRef are empty. The DcmDsdSidTabSessionLevelRef and DcmDsdSidTabSecurityLevelRef are configurable only from data identifiers (DIDs).</p> <p>DcmDsdSidTabServiceId is set to one of the following:</p> <ul style="list-style-type: none"> <li>▶ 0x10 for DIAGNOSTIC-SESSION-CONTROL.</li> <li>▶ 0x27 for DIAGNOSTIC-SECURITY-ACCESS.</li> <li>▶ 0x22 for DIAGNOSTIC-READ-DATA-BY-IDENTIFIER.</li> <li>▶ 0x2E for DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER.</li> <li>▶ 0x31 for DIAGNOSTIC-ROUTINE-CONTROL.</li> <li>▶ 0x19 for DIAGNOSTIC-READ-DTC-INFORMATION.</li> <li>▶ 0x11 for DIAGNOSTIC-ECU-RESET.</li> <li>▶ 0x28 for DIAGNOSTIC-COM-CONTROL.</li> <li>▶ 0x09 for DIAGNOSTIC-REQUEST-VEHICLE-INFO.</li> <li>▶ 0x06 for DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS.</li> <li>▶ 0x34 for DIAGNOSTIC-REQUEST-DOWNLOAD.</li> <li>▶ 0x35 for DIAGNOSTIC-REQUEST-UPLOAD.</li> <li>▶ 0x36 for DIAGNOSTIC-DATA-TRANSFER.</li> <li>▶ 0x37 for DIAGNOSTIC-REQUEST-TRANSFER-EXIT.</li> <li>▶ 0x2A for DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID.</li> <li>▶ 0x2F for DIAGNOSTIC-IO-CONTROL.</li> <li>▶ 0x3D for DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS.</li> <li>▶ 0x23 for DIAGNOSTIC-READ-MEMORY-BY-ADDRESS.</li> <li>▶ 0x85 for DIAGNOSTIC-CONTROL-DTC-SETTING.</li> <li>▶ 0x2C for DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER.</li> <li>▶ 0x14 for DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFORMATION.</li> <li>▶ 0x08 for DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE.</li> <li>▶ 0x03 for DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC.</li> <li>▶ 0x07 for DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ 0x0A for DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC-PERMANENT-STATUS.</li> <li>▶ 0x02 for DIAGNOSTIC-REQUEST-POWERTRAIN_FREEZE-FRAME-DATA.</li> <li>▶ 0x01 for DIAGNOSTIC-REQUEST-CURRENT-POWERTRAIN-DIAGNOSTIC-DATA.</li> <li>▶ 0x04 for DIAGNOSTIC-CLEAR-RESET-EMISSION-RELATED-INFO.</li> </ul> <p>DcmDsdSidTabSessionLevelRef references the following:</p> <ul style="list-style-type: none"> <li>▶ Nothing for DIAGNOSTIC-SESSION-CONTROL diagnostic service instance.</li> <li>▶ All the DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the diagnostic service instance references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</li> </ul> <p>DcmDsdSidTabSecurityLevelRef references the following:</p> <ul style="list-style-type: none"> <li>▶ Nothing for DIAGNOSTIC-SECURITY-ACCESS.</li> <li>▶ All the DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the diagnostic service instance references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</li> </ul> <p><b>Important:</b> If <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a> yields no valid DIAGNOSTIC-ACCESS-PERMISSION or a DIAGNOSTIC-ACCESS-PERMISSION that does not reference DIAGNOSTIC-SECURITY-LEVEL or DIAGNOSTIC-SESSION elements, then the corresponding DcmDsdSidTabSecurityLevelRef or DcmDsdSidTabSessionLevelRef containers for that DiagnosticServiceInstance will be empty.</p> <p>DcmDsdSidTabSubfuncAvail is set to one the following:</p> <ul style="list-style-type: none"> <li>▶ Not set for DIAGNOSTIC-SESSION-CONTROL.</li> <li>▶ Not set for DIAGNOSTIC-SECURITY-ACCESS.</li> <li>▶ Set to false for DIAGNOSTIC-READ-DATA-BY-IDENTIFIER.</li> <li>▶ Set to false for DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-ROUTINE-CONTROL</code>.</li> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-READ-DTC-INFORMATION</code>.</li> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-ECU-RESET</code>.</li> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-COM-CONTROL</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-VEHICLE-INFO</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-DOWNLOAD</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-UPLOAD</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-DATA-TRANSFER</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-TRANSFER-EXIT</code>.</li> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-CONTROL-DTC-SETTING</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID</code>.</li> <li>▶ Set to <code>true</code> for <code>DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-IO-CONTROL</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-READ-MEMORY-BY-ADDRESS</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFORMATION</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-CONTROL-OF-ON-BOARD-DEVICE</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-EMISSION-RELATED-DTC-PERMANENT-STATUS</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-CURRENT-POWERTRAIN-DIAGNOSTIC-DATA</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-REQUEST-POWERTRAIN_FREEZE-FRAME-DATA</code>.</li> <li>▶ Set to <code>false</code> for <code>DIAGNOSTIC-CLEAR-RESET-EMISSION-RELATED-INFO</code>.</li> </ul>

### 3.4.11.1. Configuration of subservices for Diagnostic Session Control

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-SESSION-CONTROL a DcmDsdSubService container is created. The container name is set to DIAGNOSTIC-SESSION/SHORT-NAME of DIAGNOSTIC-SESSION-CONTROL.</p> <p>DcmDsdSubServiceId is set to DIAGNOSTIC-SESSION/ID.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-SESSION-CONTROL references. For more information, see also <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p> <p>DcmDsdSubServiceSecurityLevelRef references all the associated DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-SESSION-CONTROL references. For more information, see also <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p>

### 3.4.11.2. Configuration of subservices for Diagnostic Security Access

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-SECURITY-ACCESS <b>two</b> DcmDsdSubService containers are created. The name for the first container is &lt;name&gt;_requestSeed, where &lt;name&gt; is set to DIAGNOSTIC-SECURITY-ACCESS/SHORT-NAME. The name for the second container is &lt;name&gt;_sendKey, where &lt;name&gt; is set to DIAGNOSTIC-SECURITY-ACCESS/SHORT-NAME.</p> <p>DcmDsdSubServiceId is set to DIAGNOSTIC-SECURITY-ACCESS/REQUEST-SEED-ID for the first DcmDsdService container. DcmDsdSubServiceId is set to DIAGNOSTIC-SECURITY-ACCESS/REQUEST-SEED-ID+1 for the second DcmDsdService container.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-SESSION entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAG-</p>

Configuration parameters	Mapping description
	<p><code>DIAGNOSTIC-SECURITY-ACCESS</code> references. For more information, see also <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p>The <code>DcmDsdSubServiceSecurityLevelRef</code> will not be configured.</p>

### 3.4.11.3. Configuration of subservices for Diagnostic Read Data By Identifier

Configuration parameters	Mapping description
<code>DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService</code>	Subservices are not applicable for the <code>DIAGNOSTIC-READ-DATA-BY-IDENTIFIER</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.4. Configuration of subservices for Diagnostic Write Data By Identifier

Configuration parameters	Mapping description
<code>DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService</code>	Subservices are not applicable for the <code>DIAGNOSTIC-WRITE-DATA-BY-IDENTIFIER</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.5. Configuration of subservices for Diagnostic IOControl

Configuration parameters	Mapping description
<code>DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService</code>	Subservices are not applicable for the <code>DIAGNOSTIC-IO-CONTROL</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

Configuration parameters	Mapping description
eTable/DcmDsdService/DcmDsdSubService	

#### 3.4.11.6. Configuration of subservices for Diagnostic Write Memory By Address

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdService-eTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the <code>DIAGNOSTIC-WRITE-MEMORY-BY-ADDRESS</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

#### 3.4.11.7. Configuration of subservices for Diagnostic Read Memory By Address

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdService-eTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the <code>DIAGNOSTIC-READ-MEMORY-BY-ADDRESS</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

#### 3.4.11.8. Configuration of subservices for Diagnostic Routine Control

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdService-eTable/DcmDsdSer-	Subservices are not applicable for the <code>DIAGNOSTIC-ROUTINE-CONTROL</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

Configuration parameters	Mapping description
vice/DcmDsdSubService	

### 3.4.11.9. Configuration of subservices for Diagnostic Read Data By Periodic Identifier

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the DIAGNOSTIC-READ-DATA-BY-PERIODIC-ID diagnostic service instance and hence DcmDsdSubService containers are not created.

### 3.4.11.10. Configuration of subservices for Diagnostic Dynamically Define Data Identifier

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER that has its Subfunction set, a DcmDsdSubService container is created. The container name is set to DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER/SUBFUNCTIONS.</p> <p>DcmDsdSubServiceId is set according to ISO-14229 to the corresponding value of DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER/SUBFUNCTIONS.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-DYNAMICALLY-DEFINE-DATA-IDENTIFIER entities with which the DIAGNOSTIC-ACCESS-PERMISSION is associated. For more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p>DcmDsdSubServiceSecurityLevelRef references all the associated DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEV-</p>



Configuration parameters	Mapping description
	EL entities with which the <code>DIAGNOSTIC-ACCESS-PERMISSION</code> is associated. For more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a> .

### 3.4.11.11. Configuration of subservices for Diagnostic Clear Diagnostic Information

Configuration parameters	Mapping description
<code>DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService</code>	Subservices are not applicable for the <code>DIAGNOSTIC-CLEAR-DIAGNOSTIC-INFORMATION</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.12. Configuration of subservices for Diagnostic Read DTC Information

Configuration parameters	Mapping description
<code>DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService</code>	<p>For every <code>DIAGNOSTIC-READ-DTC-INFORMATION</code> that has its <code>CATEGORY</code> set, a <code>DcmDsdSubService</code> container is created. The container name is set to <code>DIAGNOSTIC-READ-DTC-INFORMATION/CATEGORY</code>.</p> <p><code>DcmDsdSubServiceId</code> is set according to ISO-14229 to the corresponding value of <code>DIAGNOSTIC-READ-DTC-INFORMATION/CATEGORY</code>.</p> <p><code>DcmDsdSubServiceSessionLevelRef</code> references all the associated <code>DcmDspSession</code> containers that correspond to the <code>DIAGNOSTIC-READ-DTC-INFORMATION</code> entities which the <code>DIAGNOSTIC-ACCESS-PERMISSION</code> associated. For more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p><code>DcmDsdSubServiceSecurityLevelRef</code> references all the associated <code>DcmDspSecurity</code> containers that correspond to the <code>DIAGNOSTIC-SECURITY-LEVEL</code> entities which the <code>DIAGNOSTIC-ACCESS-PERMISSION</code> associated with the <code>DIAGNOSTIC-READ-DTC-INFORMATION</code> references. For more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p>

### 3.4.11.13. Configuration of diagnostic subservices EcuReset

Configuration parameters	Mapping description
DcmConfigSet/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-ECU-RESET that has its CATEGORY set, a DcmDsdSubService container is created. The container name is set to DIAGNOSTIC-ECU-RESET/CATEGORY.</p> <p>DcmDsdSubServiceId is set according to ISO-14229 to the corresponding value of DIAGNOSTIC-ECU-RESET/CATEGORY. In case you want to configure a custom sub service, then set the value of the sub service id based on ECU-RESET/CUSTOM-SUB-FUNCTION-NUMBER.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-ECU-RESET entities with which the DIAGNOSTIC-ACCESS-PERMISSION is associated. For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p> <p>DcmDsdSubServiceSecurityLevelRef references all the associated DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-ECU-RESET references. For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p>

### 3.4.11.14. Configuration of diagnostic subservices ComControl

Configuration parameters	Mapping description
DcmConfigSet/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-COM-CONTROL that has its CATEGORY set, a DcmDsdSubService container is created. The container name is set to DIAGNOSTIC-COM-CONTROL/CATEGORY.</p> <p>DcmDsdSubServiceId is set according to ISO-14229 to the corresponding value of DIAGNOSTIC-COM-CONTROL/CATEGORY. In case you want to configure a custom sub service, then set the value of the sub service id based on COM-CONTROL/CUSTOM-SUB-FUNCTION-NUMBER.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-COM-CONTROL entities with which the DIAGNOSTIC-ACCESS-PERMISSION is associated. For</p>

Configuration parameters	Mapping description
	<p>more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p> <p>DcmDsdSubServiceSecurityLevelRef references all the associated DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities which the DIAGNOSTIC-ACCESS-PERMISSION associated with the DIAGNOSTIC-COM-CONTROL references. For more information, see <a href="#">Section 3.4.11.22, “Determining the access permissions for diagnostic service instances”</a>.</p>

### 3.4.11.15. Configuration of subservices for Diagnostic RequestOnBoardMonitoringTestResults

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the DIAGNOSTIC-REQUEST-ON-BOARD-MONITORING-TEST-RESULTS diagnostic service instance and hence DcmDsdSubService containers are not created.

### 3.4.11.16. Configuration of subservices for Diagnostic RequestVehicleInfo

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the DIAGNOSTIC-REQUEST-VEHICLE-INFO diagnostic service instance and hence DcmDsdSubService containers are not created.

### 3.4.11.17. Configuration of subservices for Diagnostic RequestDownload

Configuration parameters	Mapping description
DcmCon- figSet/DcmDsd/ DcmDsdService- eTable/DcmDsdSer- vice/DcmDsdSubSer- vice	Subservices are not applicable for the <code>DIAGNOSTIC-REQUEST-DOWNLOAD</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.18. Configuration of subservices for Diagnostic RequestUpload

Configuration parameters	Mapping description
DcmCon- figSet/DcmDsd/ DcmDsdService- eTable/DcmDsdSer- vice/DcmDsdSubSer- vice	Subservices are not applicable for the <code>DIAGNOSTIC-REQUEST-UPLOAD</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.19. Configuration of subservices for Diagnostic DataTransfer

Configuration parameters	Mapping description
DcmCon- figSet/DcmDsd/ DcmDsdService- eTable/DcmDsdSer- vice/DcmDsdSubSer- vice	Subservices are not applicable for the <code>DIAGNOSTIC-DATA-TRANSFER</code> diagnostic service instance and hence <code>DcmDsdSubService</code> containers are not created.

### 3.4.11.20. Configuration of subservices for Diagnostic RequestTransferExit

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	Subservices are not applicable for the DIAGNOSTIC-REQUEST-TRANSFER-EXIT diagnostic service instance and hence DcmDsdSubService containers are not created.

### 3.4.11.21. Configuration of diagnostic subservices ControlDTCSetting

Configuration parameters	Mapping description
DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService	<p>For every DIAGNOSTIC-CONTROL-DTC-SETTING, a DcmDsdSubService container is created. The container name is set to DIAGNOSTIC-CONTROL-DTC-SETTING/SHORTNAME.</p> <p>DcmDsdSubServiceId is set to DIAGNOSTIC-CONTROL-DTC-SETTING/DTC-SETTING-PARAMETER.</p> <p>DcmDsdSubServiceSessionLevelRef references all the associated DcmDspSession containers that correspond to the DIAGNOSTIC-CONTROL-DTC-SETTING entities with which the DIAGNOSTIC-ACCESS-PERMISSION is associated. For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p> <p>DcmDsdSubServiceSecurityLevelRef references all the associated DcmDspSecurity containers that correspond to the DIAGNOSTIC-SECURITY-LEVEL entities with which the DIAGNOSTIC-ACCESS-PERMISSION is associated . For more information, see <a href="#">Section 3.4.11.22, "Determining the access permissions for diagnostic service instances"</a>.</p>

### 3.4.11.22. Determining the access permissions for diagnostic service instances

The access permission of a diagnostic service instance is determined as follows:

- If the DIAGNOSTIC-SESSION-CONTROL-CLASS/ACCESS-PERMISSION-VALIDITY of the diagnostic service instance is ACCESS-PERMISSION-SERVICE-INSTANCE, the access permission of the diagnostic service instance is used.

- ▶ If the `DIAGNOSTIC-SESSION-CONTROL-CLASS/ACCESS-PERMISSION-VALIDITY` of the diagnostic service instance is `ACCES-PERRMISSION-SERVICE-CLASS`, the access permission of the diagnostic service instance class is used.
- ▶ If the `DIAGNOSTIC-SESSION-CONTROL-CLASS/ACCESS-PERMISSION-VALIDITY` of the diagnostic service instance is `ACCESS-PERMISSION-INSTANCE-OVERRIDES-CLASS`, the access permission of the diagnostic service instance is used. If the access permission of the diagnostic service instance is not available, then the access permission of the diagnostic service instance class is used.

### 3.4.11.23. Determining the type for a routine signal

- ▶ `UINT8` for `BASE-TYPE-ENCODING NONE` and `BASE-TYPE-SIZE 8`
- ▶ `UINT16` for `BASE-TYPE-ENCODING NONE` and `BASE-TYPE-SIZE 16`
- ▶ `UINT32` for `BASE-TYPE-ENCODING NONE` and `BASE-TYPE-SIZE 32`
- ▶ `SINT8` for `BASE-TYPE-ENCODING 2C` and `BASE-TYPE-SIZE 8`
- ▶ `SINT16` for `BASE-TYPE-ENCODING 2C` and `BASE-TYPE-SIZE 16`
- ▶ `SINT32` for `BASE-TYPE-ENCODING 2C` and `BASE-TYPE-SIZE 32`
- ▶ `BOOLEAN` for `BASE-TYPE-ENCODING BOOLEAN` and `BASE-TYPE-SIZE 1` or `BASE-TYPE-SIZE 8`
- ▶ `VARIABLE-LENGTH` for `ARRAY-SIZE-SEMANTICS VARIABLE-SIZE`, `BASE-TYPE-ENCODING NONE`, and `BASE-TYPE-SIZE 8`

### 3.4.11.24. Configuration of `DcmProcessingConditions`

- ▶ A `DcmModeCondition` is created based on `DiagnosticEnvModeCondition` that is valid. The `DcmModeCondition` has the following parameter set: A `DcmConditionType` that has set the value of the `CompareType` attribute of `DiagnosticEnvModeCondition`, and two references `DcmBswModeRef` and `DcmSwcModeRef`. Only one reference is set at a time depending on the `ModeElement` attribute from the `DiagnosticEnvModeCondition`. The `DcmBswModeRef` has a reference to `DiagnosticEnvBswModeElement` and the `DcmSwcModeRef` has a reference to `DiagnosticEnvSwcModeElement`.
- ▶ A `DcmModeRule` is created based on the `DiagnosticEnvConditionFormula`. The `DcmModeRule` has the following parameters set: A `DcmLogicalOperator` that has set the value of the `Op` attribute of `DiagnosticEnvConditionFormula`. A `DcmModeRuleNrcValue` that has set the value of the `NrcValues` attribute of `DiagnosticEnvConditionFormula` and a `DcmArgumentRef`. The `DcmArgumentRef` has references to the parts of the `DiagnosticEnvConditionFormula`.

### 3.4.11.25. Configuration of `DcmDspPid`

- ▶ The `DcmDspPidIdentifier` is created if the id of the `DiagnosticParameterIdentifier` is valid.

- ▶ The `DcmDspPidService` has the following parameter set: A `DCM_SERVICE_01`, `DCM_SERVICE_02` if `DiagnosticParameterIdentifier` is referenced by `DiagnosticRequestCurrentPowertrainData` or `DiagnosticRequestPowertrainFreezeFrameData` services, and the value `DCM_SERVICE_01_02` if it's referenced by both services.
- ▶ The `DcmDspPidSize` has the `pidSize` parameter of the `DiagnosticParameterIdentifier` if it's valid.

#### 3.4.11.25.1. Configuration of `DcmDspPidData`

- ▶ The `DcmDspPidDataPos` value is set to `maxNumberOfElements` from the `DiagnosticDataElement` of the `DiagnosticParameterIdentifier`.
- ▶ The `DcmDspPidDataSize` value is set depending on the `maxNumberOfElements` from the `DiagnosticDataElement` of the `DiagnosticParameterIdentifier`. If it's referenced by a `DiagnosticServiceSwMapping` and has a valid `MappedFlatSwcServiceDependency` mapping that has `RoleBasedPortAssignment` that have a valid `PortPrototype` mapping then, if the `PortPrototype` has `ProvidedInterface` it's considered of type `SenderReceiverInterface` and the value will be calculated like this:  $\text{maxNumberOfElements} * (\text{baseType.getBaseTypeSize()} / 8)$  If `ProvidedInterface` is of type `ClientServerInterface` then the value will be the `maxNumberOfElements` of the `DiagnosticDataElement`.

#### 3.4.11.25.2. Configuration of `DcmDspPidDataSupportInfo`

- ▶ The `DcmDspPidDataSupportInfoBit` value is set from `supportInfoBit` value of the `DiagnosticParameterIdentifier`.
- ▶ The `DcmDspPidDataSupportInfoRef` value is a reference set to `DcmDspPidSupportInfo` based on the `supportInfoByte` value of the `DiagnosticParameterIdentifier`.

#### 3.4.11.25.3. Configuration of `DcmDspPidService01`

- ▶ The `DcmDspPidDataReadFnc` value is set based on the `DiagnosticDataElement` if it's referenced in a `DiagnosticServiceSwMapping` and has a valid `MappedBswServiceDependency` mapping with a `BswServiceDependency` that has a `RoleBasedBswModuleEntryAssignment` then the value will be the `Role` value.
- ▶ The `DcmDspPidDataUsePort` value is set depending on the `DiagnosticServiceSwMapping`. The values will be the following: `USE_DATA_SYNCH_FNC` if the `DiagnosticServiceSwMapping` has a valid `MappedBswServiceDependency` mapping or `USE_DATA_SYNCH_CLIENT_SERVER` if the `DiagnosticServiceSwMapping` has a valid `MappedFlatSwcServiceDependency` or `SwcServiceDependency` mapping.

#### 3.4.11.25.4. Configuration of DcmDspPidSupportInfo

- ▶ The `DcmDspPidSupportInfoLen` value is set to `Size` from the parameter `DiagnosticSupportInfoByte` of the `DiagnosticParameterIdentifier`.
- ▶ The `DcmDspPidSupportInfoPos` value is set to `Position` from the parameter `DiagnosticSupportInfoByte` of the `DiagnosticParameterIdentifier`.

### 3.4.12. Dem

Configuration parameters	Mapping description
<code>DemConfigSet</code>	Check if a <code>DemConfigSet</code> container is created inside the module. If one already exists then an additional one is not created, otherwise a new <code>DemConfigSet</code> container is created.
<code>DemConfigSet/DemDTCClass</code>	<p>For every <code>DIAGNOSTIC-TROUBLE-CODE-UDS</code> that is assigned to the configured ECU-INSTANCE one <code>DemDTCClass</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the mangled name of the <code>DIAGNOSTIC-TROUBLE-CODE-UDS</code>.</p> <p><code>DemUdsDTC</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-UDS/UDS-DTC-VALUE</code>.</p> <p><code>DemDTCFunctionalUnit</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-UDS/FUNCTIONAL-UNIT</code>.</p> <p><code>DemDTCSeverity</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-UDS/SEVERITY</code>.</p> <p><code>DemImmediateNvStorage</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-UDS/DIAGNOSTIC-TROUBLE-CODE-PROPS/IMMEDIATE-NV-DATA-STORAGE</code>.</p> <p><code>DemObdDTC</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-OBDOBD-DTC-VALUE</code>.</p>
<code>DemGeneral/DemGroupOfDTC</code>	<p>For every <code>DIAGNOSTIC-TROUBLE-CODE-GROUP</code> that is assigned to the configured ECU-INSTANCE one <code>DemGroupOfDTC</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the mangled name of the <code>DIAGNOSTIC-TROUBLE-CODE-GROUP</code>.</p> <p><code>DemGroupDTCs</code> is set to <code>DIAGNOSTIC-TROUBLE-CODE-GROUP/GROUP-NUMBER</code>.</p>
<code>DemGeneral/DemEnableCondition</code>	For every <code>DIAGNOSTIC-ENABLE-CONDITION</code> that is assigned to the configured ECU-INSTANCE one <code>DemEnableCondition</code> container is created. The container name is <code>&lt;name&gt;</code> , where <code>&lt;name&gt;</code> is the mangled name of the <code>DIAGNOSTIC-ENABLE-CONDITION</code> .



Configuration parameters	Mapping description
	DemEnableConditionStatus is set to DIAGNOSTIC-ENABLE-CONDITION/INIT-VALUE.
DemGeneral/DemEnableConditionGroup	<p>For every different DIAGNOSTIC-ENABLE-CONDITION-GROUP that contains different sets of DIAGNOSTIC-ENABLE-CONDITION references, that is assigned to the configured ECU-INSTANCE, a single DemEnableConditionGroup container is created. The container name is &lt;name&gt;, where &lt;name&gt; is the mangled name of the DIAGNOSTIC-ENABLE-CONDITION-GROUP.</p> <p>DemEnableConditionRef references all the associated DemEnableCondition containers that correspond to the DIAGNOSTIC-ENABLE-CONDITION elements aggregated within this DIAGNOSTIC-ENABLE-CONDITION-GROUP.</p>
DemGeneral/DemOperationCycle	<p>A DemOperationCycle container is created for every DIAGNOSTIC-OPERATION-CYCLE that is assigned to the configured ECU-INSTANCE and which is used as either a DIAGNOSTIC-OPERATION-CYCLE-REF or as a HEALING-CYCLE attribute for a DIAGNOSTIC-CONNECTED-INDICATOR. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the mangled name of the DIAGNOSTIC-OPERATION-CYCLE.</p> <p>DemOperationCycleAutomaticEnd is set to DIAGNOSTIC-OPERATION-CYCLE/AUTOMATIC-END.</p> <p>DemOperationCycleType is set to DIAGNOSTIC-OPERATION-CYCLE/TYPE.</p>
DemGeneral/DemAgingCycle	<p>A DemAgingCycle container is created for every DIAGNOSTIC-OPERATION-CYCLE that is assigned to the configured ECU-INSTANCE and which is used as either a DIAGNOSTIC-OPERATION-CYCLE-REF or as AGING-CYCLE for a DIAGNOSTIC-AGING. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;PREFIX&gt; is DEM_AG_CYCLE_ and &lt;name&gt; is the mangled name of the DIAGNOSTIC-OPERATION-CYCLE.</p>
DemGeneral	<p>Check if a DemGeneral container is created inside the module. If a container already exists, no additional container is created. Otherwise, a new DemGeneral container is created.</p> <p>DemEnableConditionSupport is set to true if at least one DIAGNOSTIC-ENABLE-CONDITION exists that is assigned to the configured ECU-INSTANCE.</p> <p>DemTypeOfFreezeFrameRecordNumeration is set to DEM_FF_REC_NUM_CONFIGURED if at least one DIAGNOSTIC-TROUBLE-CODE-PROPS exists that is assigned to the configured ECU-INSTANCE and contains at least one DiagnosticFreezeFrame with a valid RecordNumber parameter.</p>

Configuration parameters	Mapping description
	<p><code>DemDebounceCounterBasedSupport</code> is set to true if the <code>DIAG-NOSTIC-EVENT</code> that is assigned to the configured <code>ECU-INSTANCE</code> has support for <code>DIAG-EVENT-DEBOUNCE-COUNTER-BASED</code>.</p> <p><code>DemDebounceTimeBasedSupport</code> is set to true if the <code>DIAGNOSTIC-EVENT</code> that is assigned to the configured <code>ECU-INSTANCE</code> has support for <code>DIAG-EVENT-DEBOUNCE-TIME-BASED</code>.</p> <p>If at least one <code>DIAGNOSTIC-COMMON-ELEMENT</code> exists that is an instance of <code>DIAGNOSTIC-ECU-INSTANCE-PROPS</code> with <code>DTC-STATUS-AVAILABILITY-MASK</code> set and is assigned to the configured <code>ECU-INSTANCE</code>, then the value of the <code>DemDtcStatusAvailabilityMask</code> attribute is configured based on the <code>DTC-STATUS-AVAILABILITY-MASK</code> from the first <code>DIAGNOSTIC-ECU-INSTANCE-PROPS</code> configured.</p> <p>If at least one <code>DIAGNOSTIC-COMMON-ELEMENT</code> exists that is an instance of <code>DIAGNOSTIC-ECU-INSTANCE-PROPS</code> with <code>OBD-SUPPORT</code> set with a different value than <code>NO-OBD-SUPPORT</code> and is assigned to the configured <code>ECU-INSTANCE</code>, then the value of the <code>DemOBDSupport</code> attribute is set to true and the <code>DemOBDSupportKind</code> attribute is configured based on the <code>OBD-SUPPORT</code> from the first <code>DIAGNOSTIC-ECU-INSTANCE-PROPS</code> configured. Otherwise the <code>DemOBDSupport</code> attribute is set to false and the <code>DemOBDSupportKind</code> is disabled.</p>
<code>DemGeneral/DemIndicator</code>	<p>For every <code>DIAGNOSTIC-INDICATOR</code> that is assigned to the configured <code>ECU-INSTANCE</code> one <code>DemIndicator</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the mangled name of the <code>DIAGNOSTIC-INDICATOR</code>.</p> <p><code>DemIndicatorID</code> is set to an auto-calculated value which specifies the index of this container.</p>
<code>DemGeneral/DemClient</code>	<p>If any <code>DiagnosticServiceInstance</code> of a <code>DiagnosticServiceTable</code> of the configured <code>EcuInstance</code> depends on <code>Dem</code>, one or two <code>DemClient</code> containers are created. The container names are <code>DemClient&lt;ProtocolType&gt;</code>, where <code>&lt;ProtocolType&gt;</code> is <code>OBD</code> and/or <code>UDS</code>.</p>
<code>DemGeneral/DemDataElementClass</code>	<p>A <code>DemDataElementClass</code> choice container is created for all the <code>DiagnosticDataElements</code> aggregated by a <code>DiagnosticTroubleCodeProps</code>, either through the <code>DiagnosticExtendedDataRecords</code> or through the <code>FreezeFrameContent</code>. For each of the <code>DiagnosticDataElements</code> linked inside a <code>DiagnosticDemProvidedDataMapping</code> element assigned to the configured <code>EcuInstance</code> a <code>DemInternalDataElementClass</code> choice container is created. The container name is</p>

Configuration parameters	Mapping description
	<p>&lt;PREFIX&gt;&lt;name&gt;, where &lt;PREFIX&gt; is DemInternalDataElementClass_ and &lt;name&gt; is the mangled shortName of the DiagnosticDataElement.</p> <p>DemInternalDataElement is set to the DIAGNOSTIC-DATA-PROVIDER category only if the values of DataProvider are reserved by the AUTOSAR standard and are one of the following:</p> <ul style="list-style-type: none"> <li>▶ DEM_AGINGCTR_DOWNCNT</li> <li>▶ DEM_AGINGCTR_UPCNT</li> <li>▶ DEM_CURRENT_FDC</li> <li>▶ DEM_CYCLES_SINCE_FIRST_FAILED</li> <li>▶ DEM_CYCLES_SINCE_LAST_FAILED</li> <li>▶ DEM_OCCCTR</li> <li>▶ DEM_OVFLIND</li> <li>▶ DEM_SIGNIFICANCE</li> </ul> <p>If DIAGNOSTIC-DATA-PROVIDER has a different value than the ones reserved by the AUTOSAR standard, then DemInternalDataElement is set to the default value DEM_AGINGCTR and a warning is issued.</p> <p>DemDataElementDataSize is set to (DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * DIAGNOSTIC-DATA-ELEMENT/SW-DATA-DEF-PROPS/SW-DATA-DEF-PROPS-CONDITIONAL/SW-BASE-TYPE/BASE-TYPE-SIZE)/8.</p> <p>By default if the DiagnosticDataElement is not linked inside a DiagnosticDem-ProvidedDataMapping then a DemExternalCSDDataElementClass choice container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;PREFIX&gt; is DemExternalCSDDataElementClass_ and &lt;name&gt; is the mangled shortName of the DiagnosticDataElement.</p> <p>DemDataElementDataSize is set to (DIAGNOSTIC-DATA-ELEMENT/MAX-NUMBER-OF-ELEMENTS * DIAGNOSTIC-DATA-ELEMENT/SW-DATA-DEF-PROPS/SW-DATA-DEF-PROPS-CONDITIONAL/SW-BASE-TYPE/BASE-TYPE-SIZE)/8.</p>
DemGeneral/DemDem-DidClass	<p>For every DIAGNOSTIC-DATA-IDENTIFIER that is assigned to the configured ECU-INSTANCE one DemDemDidClass container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the mangled name of the DIAGNOSTIC-DATA-IDENTIFIER and the &lt;PREFIX&gt; is DemDidClass_.</p>

Configuration parameters	Mapping description
	<p><b>DemDidIdentifier</b> is set to DIAGNOSTIC-DATA-IDENTIFIER/ID.</p> <p><b>DemDidDataElementClassRef</b> references all the associated <b>DemDataElementClass</b> containers that correspond to the DIAGNOSTIC-DATA-ELEMENT elements aggregated within this DIAGNOSTIC-DATA-IDENTIFIER.</p>
DemGeneral/DemFreezeFrameClass	<p>For every different set of DIAGNOSTIC-DATA-IDENTIFIER elements from DIAGNOSTIC-DATA-IDENTIFIER-SET aggregated by each DIAGNOSTIC-TROUBLE-CODE-PROPS that is assigned to the configured ECU-INSTANCE, only one <b>DemFreezeFrameClass</b> container is created. The container name is &lt;name&gt;&lt;SUFFIX&gt;, where &lt;name&gt; is <b>DiagnosticFreezeFrameClass_</b> and the &lt;SUFFIX&gt; is the number of the <b>DemFreezeFrameClass</b> containers created.</p> <p><b>DemDidClassRef</b> references all the associated <b>DiagnosticDataIdentifiers</b> elements aggregated within a DIAGNOSTIC-DATA-IDENTIFIER-SET.</p>
DemGeneral/DemFreezeFrameRecNumClass	<p>For every different set of DIAGNOSTIC-FREEZE-FRAME-RECORD-NUMBER elements aggregated by each DIAGNOSTIC-TROUBLE-CODE-PROPS that is assigned to the configured ECU-INSTANCE only one <b>DemFreezeFrameRecNumClass</b> container is created. The container name is &lt;name&gt;&lt;SUFFIX&gt;, where &lt;name&gt; is <b>DiagnosticFreezeFrameRecNumClass_</b> and the &lt;SUFFIX&gt; is the number of the <b>DemFreezeFrameRecNumClass</b> containers created.</p> <p><b>DemFreezeFrameRecordNumber</b> is set to the DIAGNOSTIC-FREEZE-FRAME-RECORD-NUMBER aggregated by a DIAGNOSTIC-TROUBLE-CODE-PROPS.</p>
DemConfigSet/DemEventParameter	<p>For each DIAGNOSTIC-EVENT that is assigned to the configured ECU-INSTANCE, one <b>DemEventParameter</b> container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the mangled name of the DIAGNOSTIC-EVENT.</p> <p><b>DemEventId</b> is set to an auto-calculated value which specified the index of this container.</p> <p><b>DemEventKind</b> is set to DIAGNOSTIC-EVENT/EVENT-KIND.</p> <p><b>DemDTCClassRef</b> references the associated <b>DemDTCClass</b> container that corresponds to the DIAGNOSTIC-TROUBLE-CODE-UDS which is mapped to the DIAGNOSTIC-EVENT via DIAGNOSTIC-EVENT-TO-TROUBLE-CODE-UDS-MAPPING.</p> <p>If a <b>DemExtendedDataClass</b> container has been created for the DIAGNOSTIC-TROUBLE-CODE-PROPS element of this DIAGNOSTIC-EVENT, De-</p>

Configuration parameters	Mapping description
	<p><code>mExtendedDataClassRef</code> is configured to refer to that <code>DemExtendedDataClass</code> container.</p> <p><code>DemFreezeFrameClassRef</code> references the associated <code>DemFreezeFrameClass</code> container that corresponds to the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/FREEZE-FRAME-CONTENT-REF</code>, where <code>DIAGNOSTIC-TROUBLE-CODE-PROPS</code> can be obtained via <code>DIAGNOSTIC-EVENT-TO-TROUBLE-CODE-UDS-MAPPING</code>.</p> <p><code>DemFreezeFrameRecNumClassRef</code> references the associated <code>DemDTCClass</code> container that corresponds to the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/DIAGNOSTIC-EXTENDED-DATA-RECORD-REF-CONDITIONAL</code>, where <code>DIAGNOSTIC-TROUBLE-CODE-PROPS</code> can be obtained via <code>DIAGNOSTIC-EVENT-TO-TROUBLE-CODE-UDS-MAPPING</code>.</p> <p><code>DemMaxNumberFreezeFrameRecords</code> is set to the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/MAX-NUMBER-FREEZE-FRAME-RECORDS</code>, where <code>DIAGNOSTIC-TROUBLE-CODE-PROPS</code> can be obtained via <code>DIAGNOSTIC-EVENT-TO-TROUBLE-CODE-UDS-MAPPING</code>.</p>
DemConfigSet/DemEventParameter/DemCallbackClearEventAllowed	<p>A <code>DemCallbackClearEventAllowed</code> container is created under <code>DemEventParameter</code> if <code>DIAGNOSTIC-EVENT/EVENT-CLEAR-ALLOWED</code> is set. The <code>&lt;name&gt;</code> is set to <code>DemCallbackClearEventAllowed</code>.</p>
DemConfigSet/DemEventParameter/DemEventClass	<p>A <code>DemEventClass</code> container is created under <code>DemEventParameter</code>. The <code>&lt;name&gt;</code> is set to <code>DemEventClass</code>.</p> <p><code>DemEventSignificance</code> is set to the <code>DIAGNOSTIC-TROUBLE-CODE-UDS/DIAGNOSTIC-TROUBLE-CODE-PROPS/SIGNIFICANCE</code> linked to the <code>DiagnosticEvent</code> if the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/SIGNIFICANCE</code> is valid. Otherwise, <code>DemEventSignificance</code> is set to the <code>DIAGNOSTIC-TROUBLE-CODE-OBDDIAGNOSTIC-TROUBLE-CODE-PROPS/SIGNIFICANCE</code> linked to the <code>DiagnosticEvent</code> via the <code>DIAGNOSTIC-TROUBLE-CODE-UDS</code> if the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/SIGNIFICANCE</code> is valid.</p> <p><code>DemEventDestination</code> is set to the first <code>MEMORY-DESTINATION-REF</code> from the <code>DIAGNOSTIC-TROUBLE-CODE-UDSDIAGNOSTIC-TROUBLE-CODE-PROPS/MEMORY-DESTINATION-REFS</code> linked to the <code>DiagnosticEvent</code> if the <code>DIAGNOSTIC-TROUBLE-CODE-PROPS/MEMORY-DESTINATION-REF</code> is valid. Otherwise, <code>DemEventDestination</code> is set to the first <code>MEMORY-DESTINATION-REF</code> from the <code>DIAGNOSTIC-TROUBLE-CODE-OBDDIAGNOSTIC-TROUBLE-CODE-PROPS/MEMORY-DESTINATION-REFS</code> linked to the</p>

Configuration parameters	Mapping description
	<p><b>DiagnosticEvent</b> <b>via the</b> DIAGNOSTIC-TROUBLE-CODE-UDS <b>if the</b> DIAGNOSTIC-TROUBLE-CODE-PROPS/MEMORY-DESTINATION-REF <b>is valid.</b></p> <p><b>DemFFPrestorageSupported</b> <b>is set to</b> DIAGNOSTIC-EVENT/PRESTORAGE-FREEZE-FRAME.</p> <p><b>DemEventPriority</b> <b>is set to</b> DIAGNOSTIC-TROUBLE-CODE-PROPS/PRIORITY.</p> <p><b>DemAgingAllowed</b> <b>is set to</b> DIAGNOSTIC-EVENT/AGING-ALLOWED.</p> <p><b>DemEventFailureCycleCounterThreshold</b> <b>is set to</b> DIAGNOSTIC-EVENT/EVENT-FAILURE-CYCLE-COUNTER-THRESHOLD.</p> <p><b>DemEnableConditionGroupRef</b> <b>references the associated</b> DemEnableConditionGroup <b>container that corresponds to the</b> DIAGNOSTIC-ENABLE-CONDITION-GROUP <b>which is mapped to the</b> DIAGNOSTIC-EVENT <b>via</b> DIAGNOSTIC-EVENT-TO-ENABLE-CONDITION-GROUP-MAPPING.</p> <p><b>DemOperationCycleRef</b> <b>references the associated</b> DemOperationCycle <b>container that corresponds to the</b> DIAGNOSTIC-OPERATION-CYCLE <b>which is mapped to the</b> DIAGNOSTIC-EVENT <b>via</b> DIAGNOSTIC-EVENT-TO-OPERATION-CYCLE-MAPPING.</p> <p><b>DemAgingCycleRef</b> <b>references the associated</b> DemAgingCycle <b>container that corresponds to the</b> DIAGNOSTIC-AGING <b>which is mapped to the</b> DIAGNOSTIC-EVENT <b>via</b> DIAGNOSTIC-EVENT-TO-TROUBLE-CODE-UDS-MAPPING.</p> <p><b>DemAgingCycleCounterThreshold</b> <b>is set to</b> DIAGNOSTIC-AGING/THRESHOLD <b>which is mapped to</b> DemAgingCycle <b>referred above.</b></p> <p><b>DemConsiderPtoStatus</b> <b>is set to</b> DIAGNOSTIC-TROUBLE-CODE-OBD/CONSIDER-PTO-STATUS.</p> <p><b>DemEventOBDRadinessGroup</b> <b>is set to</b> DIAGNOSTIC-TROUBLE-CODE-OBD/EVENT-OBD-READINESS-GROUP.</p>
DemConfigSet/DemEventParameter/DemEventClass/DemIndicatorAttribute	<p><b>The</b> DemIndicatorAttribute <b>container is created for every</b> DIAGNOSTIC-CONNECTED-INDICATOR <b>aggregated by the</b> DIAGNOSTIC-EVENT.</p> <p><b>The container name is</b> &lt;PREFIX&gt;&lt;name&gt;, <b>where</b> &lt;name&gt; <b>is the mangled name of the</b> DIAGNOSTIC-CONNECTED-INDICATOR.</p>

Configuration parameters	Mapping description
	<p>DemIndicatorBehaviour <b>is set to</b> DIAGNOSTIC-CONNECTED-INDICATOR/BEHAVIOR.</p> <p>DemIndicatorHealingCycleRef <b>references the associated</b> DemOperationCycle <b>container that corresponds to</b> DIAGNOSTIC-CONNECTED-INDICATOR/HEALING-CYCLE-REF.</p> <p>DemIndicatorRef <b>references the associated</b> DemIndicator <b>container that corresponds to</b> DIAGNOSTIC-CONNECTED-INDICATOR/INDICATOR-REF.</p> <p>DemIndicatorHealingCycleCounterThreshold <b>is set to</b> DIAGNOSTIC-CONNECTED-INDICATOR/INDICATOR-REF/HEALING-CYCLE-COUNTER-THRESHOLD.</p>
DemConfigSet/DemEventParameter/DemEventClass/DemDebounceAlgorithmClass	<p><b>A</b> DemDebounceAlgorithmClass <b>container is created for every</b> DemEventClass <b>based on the</b> DiagnosticDebounceAlgorithmProps <b>mapped to the</b> DiagnosticEvent <b>via</b> DiagnosticEventToDebounceAlgorithmMapping. <b>The</b> DemDebounceAlgorithmClass <b>can be one of the following types:</b></p> <ul style="list-style-type: none"> <li>▶ DemDebounceMonitorInternal</li> <li>▶ DemDebounceCounterBased</li> <li>▶ DemDebounceTimeBase</li> </ul>
DemConfigSet/DemEventParameter/DemEventClass/DemDebounceAlgorithmClass/DemDebounceTimeBase	<p>DemDebounceTimeFailedThreshold <b>is set to</b> DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-TIME-BASED/TIME-FAILED-THRESHOLD.</p> <p>DemDebounceTimePassedThreshold <b>is set to</b> DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-TIME-BASED/TIME-PASSED-THRESHOLD.</p>
DemConfigSet/DemEventParameter/DemEventClass/DemDebounceAlgorithmClass/DiagEventDebounceCounterBased	<p>DemDebounceCounterDecrementStepSize <b>is set to</b> DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-DECREMENT-STEP-SIZE.</p> <p>DemDebounceCounterFailedThreshold <b>is set to</b> DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-FAILED-THRESHOLD.</p> <p>DemDebounceCounterIncrementStepSize <b>is set to</b> DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-INCREMENT-STEP-SIZE.</p>

Configuration parameters	Mapping description
	<p>DemDebounceCounterJumpDown is set to DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-JUMP-DOWN.</p> <p>DemDebounceCounterJumpDownValue is set to DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-JUMP-DOWN-VALUE.</p> <p>DemDebounceCounterJumpUp is set to DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-JUMP-UP.</p> <p>DemDebounceCounterJumpUpValue is set to DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-JUMP-UP-VALUE.</p> <p>DemDebounceCounterPassedThreshold is set to DIAGNOSTIC-DEBOUNCE-ALGORITHM-PROPS/DIAG-EVENT-DEBOUNCE-COUNTER-BASED/COUNTER-PASSED-THRESHOLD.</p>
DemGeneral/DemExtendedDataClass	<p>For every distinct and non-empty set of DIAGNOSTIC-EXTENDED-DATA-RECORD elements that is referenced by any DIAGNOSTIC-TROUBLE-CODE-PROPS element of the configured ECU-INSTANCE, a single DemExtendedDataClass container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the mangled name of one of the DIAGNOSTIC-TROUBLE-CODE-PROPS elements referencing that list.</p> <p>DemExtendedDataRecordClassRef references all the associated DemExtendedDataRecordClass containers that correspond to the DIAGNOSTIC-EXTENDED-DATA-RECORD aggregated within the DIAGNOSTIC-TROUBLE-CODE-PROPS.</p>
DemGeneral/DemExtendedDataRecordClass	<p>For every DIAGNOSTIC-EXTENDED-DATA-RECORD that is assigned to the configured ECU-INSTANCE one DemExtendedDataRecordClass container is created. The container &lt;name&gt; is the mangled name of the DIAGNOSTIC-EXTENDED-DATA-RECORD.</p> <p>DemExtendedDataRecordNumber is set to DIAGNOSTIC-EXTENDED-DATA-RECORD/RECORD-NUMBER.</p> <p>DemExtendedDataRecordUpdate is set to DIAGNOSTIC-EXTENDED-DATA-RECORD/UPDATE.</p>



Configuration parameters	Mapping description
	DemDataElementClassRef references all the associated DemDataElement-Class containers that correspond to the DIAGNOSTIC-DATA-ELEMENT elements aggregated within this DIAGNOSTIC-EXTENDED-DATA-RECORD.

### 3.4.13. DoIP

Configuration parameters	Mapping description
DoIPConfigSet	<p>One DoIPConfigSet container and one DoIPConnections subcontainer are created if the imported ECU-INSTANCE sends and/or receives one or more PDUs that belong to a DO-IP-TP-CONNECTION. A PDU belongs to a DO-IP-TP-CONNECTION if the DO-IP-TP-CONNECTION references the PDU via TP-SDU-REF.</p> <p>The VLAN of a DO-IP-TP-CONNECTION is the PHYSICAL-CHANNEL of the PDU-TRIGGERING that the DO-IP-TP-CONNECTION references via TP-SDU-REF.</p> <p>If the ECU-INSTANCE sends a PDU via a DO-IP-TP-CONNECTION, the <i>local diagnosis address</i> and the <i>tester address</i> of the DO-IP-TP-CONNECTION are the ADDRESS values of the DO-IP-LOGIC-ADDRESS elements that the DO-IP-TP-CONNECTION references via DO-IP-SOURCE-ADDRESS, respectively via DO-IP-TARGET-ADDRESS.</p> <p>If the ECU-INSTANCE receives a PDU via a DO-IP-TP-CONNECTION, the <i>local diagnosis address</i> and the <i>tester address</i> of the DO-IP-TP-CONNECTION are the ADDRESS values of the DO-IP-LOGIC-ADDRESS elements that the DO-IP-TP-CONNECTION references via DO-IP-TARGET-ADDRESS, respectively via DO-IP-SOURCE-ADDRESS.</p> <p>A <i>DoIP channel</i> consists of the set of all DO-IP-TP-CONNECTION elements via which the ECU-INSTANCE:</p> <ul style="list-style-type: none"> <li>▶ sends DCM-I-PDU elements that have their DIAG-PDU-TYPE either set to DIAG-RESPONSE or not set at all;</li> <li>▶ receives DCM-I-PDU elements that have their DIAG-PDU-TYPE either set to DIAG-REQUEST or not set at all.</li> </ul>

Configuration parameters	Mapping description
	Moreover, the <code>DO-IP-TP-CONNECTION</code> elements have to share the same <i>VLAN</i> , <i>local diagnosis address</i> , and <i>tester address</i> in order to belong to the same <i>DoIP channel</i> .
DoIPConfigSet/DoIPConnections/DoIPTcpConnection, DoIPConfigSet/DoIPConnections/DoIPUdpVehicleAnnouncement, DoIPConfigSet/DoIPConnections/DoIPUdpConnection	<p>If the imported <code>ECU-INSTANCE</code> sends and/or receives one or more PDUs of type <code>GENERAL-PURPOSE-PDU</code> with <code>CATEGORY</code> set to <code>DoIP</code> via one or several <i>local</i> <code>SOCKET-ADDRESS</code> elements, one or more subcontainers are added to <code>DoIPConfigSet/DoIPConnections</code>. If the PDUs are sent and/or received via <code>Tcp</code> <code>SOCKET-ADDRESS</code> elements, one <code>DoIPTcpConnection</code> container is created. For more information about <code>SOCKET-ADDRESS</code>, see <a href="#">Section 3.4.43, “SoAd”</a>. The container name is <code>&lt;chn&gt;_&lt;remote_addr&gt;Tcp</code>, where <code>&lt;chn&gt;</code> is the mangled name of the <code>ETHERNET-PHYSICAL-CHANNEL</code> the PDUs are transmitted on and <code>&lt;remote_addr&gt;</code> is the mangled name of the <i>remote</i> <code>SOCKET-ADDRESS</code>. If there are two or more <i>local</i> <code>Tcp</code> <code>SOCKET-ADDRESS</code> elements that exchange <code>DoIP</code> PDUs using the same <i>remote</i> <code>SOCKET-ADDRESS</code>, multiple <code>DoIPTcpConnection</code> containers are created. A suffix is appended to the container names of the second, third, etc. of these containers to ensure unique container names.</p> <p>If there is only one PDU sent to the IPv4-limited broadcast address (255.-255.255.255) or to the IPv6-limited broadcast address (FF02:0:0:0:0:0:1) via one <i>local</i> <code>Udp</code> <code>SOCKET-ADDRESS</code>, a <code>DoIPUdpVehicleAnnouncement</code> container is created. The container name is <code>&lt;chn&gt;_&lt;remote_addr&gt;UdpVehicleAnnouncement</code>.</p> <p>If there are PDUs sent and/or received via a <i>local</i> <code>Udp</code> <code>SOCKET-ADDRESS</code>, the PDUs are grouped according to the <i>remote</i> <code>SOCKET-ADDRESS</code> they are sent to or received from. If there are more than two PDUs for a given <i>remote</i> address, these PDUs are further grouped according to the <i>local</i> <code>SOCKET-ADDRESS</code> they belong to. For each of the resulting groups, one <code>DoIPUdpConnection</code> container is created. The container name is <code>&lt;chn&gt;_&lt;remote_addr&gt;Udp</code>. A suffix is appended to the container names of the second, third, etc. of these containers to ensure unique container names.</p>
DoIPConfigSet/DoIPConnections/DoIPTcpConnection/DoIPSoAdRxPdu, DoIPConfigSet/DoIPConnections/DoIPUdpConnection/DoIPSoAdRxPdu	If received PDUs exist for which the container <code>DoIPTcpConnection</code> or <code>DoIPUdpConnection</code> has been created, one <code>DoIPSoAdRxPdu</code> subcontainer is added. <code>DoIPSoAdRxPduRef</code> references the representation of the first received PDU in the <code>EcuC</code> module. A warning is issued if multiple received PDUs were found for the <code>DoIPTcpConnection/DoIPUdpConnection</code> container.

Configuration parameters	Mapping description
DoIPConfigSet/DoIPConnections/DoIPTcpConnection/DoIPSoAdTxPdu, DoIPConfigSet/DoIPConnections/DoIPUdpConnection/DoIPSoAdTxPdu	If sent PDUs exist for which the container DoIPTcpConnection or DoIPUdpConnection has been created, one DoIPSoAdTxPdu subcontainer is added. DoIPSoAdTxPduRef references the representation of the first sent PDU in the EcuC module. A warning is issued if multiple sent PDUs were found for the DoIPTcpConnection/DoIPUdpConnection container.
DoIPConfigSet/DoIPConnections/DoIPUdpVehicleAnnouncement/DoIPSoAdTxPdu	If sent PDUs exist for which the container DoIPUdpVehicleAnnouncement has been created, one DoIPSoAdTxPdu sub container is added. DoIPSoAdTxPduRef references the representation of the first sent PDU in the EcuC module. A warning is issued if multiple sent PDUs were found for the DoIPUdpVehicleAnnouncement container.
DoIPConfigSet/DoIPConnections/DoIPTargetAddress	One DoIPTargetAddress container is created for each valid and unique <i>local diagnosis address</i> value of any DO-IP-TP-CONNECTION of the ECU-INSTANCE. The container name is DoIPTargetAddress_<address>, where <address> is the value of the <i>local diagnosis address</i> . This value is also configured in DoIPTargetAddressValue.
DoIPConfigSet/DoIPTester	One DoIPTester container is created for each valid and unique <i>tester address</i> value. The container name is DoIPTester_<address>, where <address> is the value of the <i>tester address</i> . This value is also configured in DoIPTesterSA.
DoIPConfigSet/DoIPChannel	<p>For each <i>DoIP channel</i>, one DoIPChannel container is created. If the <i>DoIP channel</i> contains one or more DO-IP-TP-CONNECTION elements that receive a PDU, the container name is the mangled name of the first of these elements, otherwise the name is the mangled name of the first contained DO-IP-TP-CONNECTION that sends a PDU.</p> <p>DoIPChannelSARef is set up to reference the DoIPConfigSet/DoIPTester container that has been created for the common <i>tester address</i> of the DO-IP-TP-CONNECTION elements of the <i>DoIP channel</i>.</p> <p>DoIPChannelTARef is set up to reference the DoIPConfigSet/DoIPConnections/DoIPTargetAddress container that has been created for the common <i>local diagnosis address</i> of the DO-IP-TP-CONNECTION elements of the <i>DoIP channel</i>.</p>
DoIPConfigSet/DoIPChannel/DoIPPduRRxPdu	If one or more of the DO-IP-TP-CONNECTION elements of the <i>DoIP channel</i> receive a PDU, one DoIPPduRRxPdu subcontainer is created. Its DoIPPduRRxPduRef references the representation of the first of these received PDUs in the

Configuration parameters	Mapping description
	<b>EcuC module.</b> A warning is issued if none of the <code>DO-IP-TP-CONNECTION</code> elements of the <i>DoIP channel</i> receives a PDU.
<code>DoIPConfigSet/DoIPChannel/DoIPPduRTxPdu</code>	If one or more of the <code>DO-IP-TP-CONNECTION</code> elements of the <i>DoIP channel</i> send a PDU, one <code>DoIPPduRTxPdu</code> subcontainer is created. Its <code>DoIPPduRTxPduRef</code> references the representation of the first of these sent PDUs in the <b>EcuC</b> module.

### 3.4.14. EcuC

Configuration parameters	Mapping description
<code>EcucPduCollection/Pdu</code>	<p>For every PDU instance (see <a href="#">Section 3.3.1.3, “Instance handling”</a>) sent, received or routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) by the imported <code>ECU-INSTANCE</code>, a <code>Pdu</code> container is created. This also includes:</p> <ul style="list-style-type: none"> <li>▶ PDU instances that represent PDUs referenced in <code>DYNAMIC-PARTS</code> or <code>STATIC-PARTS</code> of <code>MULTIPLEXED-I-PDU</code> elements.</li> <li>▶ PDU instances that represent PDUs that are transmitted within <code>CONTAINER-I-PDU</code> elements. For information about <code>CONTAINER-I-PDU</code>, see <a href="#">Section 3.4.29, “lpduM”</a>).</li> </ul> <p>The container name is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the PDU belonging to the PDU instance.</p> <p>For every PDU referenced by a <code>TP-CONNECTION</code> belonging to the imported <code>ECU-INSTANCE</code>, a <code>Pdu</code> container is created as well. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the PDU.</p> <p>If the <code>ECU-INSTANCE</code> receives multiple <code>NM-PDU</code> elements in one <code>CAN-CLUSTER</code> or <code>J-1939-CLUSTER</code>, only one <code>Pdu</code> container is created representing the PDU with the lowest <code>CAN-ID</code>.</p> <p>For an <code>NM-PDU</code> an <code>NmUserDataPdu</code> container is created if at least one of the following conditions is met:</p> <ul style="list-style-type: none"> <li>▶ The <code>NM-PDU</code> contains an <code>I-SIGNAL-TO-I-PDU-MAPPING</code> that references an <code>I-SIGNAL</code> or an <code>I-SIGNAL-GROUP</code>.</li> <li>▶ The <code>NM-PDU</code> is sent and belongs to a <i>PNC-enabled</i> <code>NM-CLUSTER</code>.</li> </ul> <p>An <code>NM-CLUSTER</code> is considered PNC-enabled in one of the following cases:</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ The NM-CLUSTER has its NM-PNC-PARTICIPATION not set to false</li> <li>▶ A PNC-MAPPING references an I-SIGNAL-I-PDU-GROUP which contains PDUs - either directly or via one of its subordinate I-SIGNAL-I-PDU-GROUP elements - that belong to the COMMUNICATION-CLUSTER referenced by the NM-CLUSTER. The referenced I-SIGNAL-I-PDU-GROUP or any of its subordinate I-SIGNAL-I-PDU-GROUP elements must also be referenced by ASSOCIATED-COM-I-PDU-GROUP-REFS of the imported ECU-INSTANCE</li> <li>▶ A PNC-MAPPING references the ETHERNET-PHYSICAL-CHANNEL which is referenced from the NM-CLUSTER as well</li> <li>▶ A PNC-MAPPING references a CAN-PHYSICAL-CHANNEL or a FLEXRAY-PHYSICAL-CHANNEL which is contained in a CAN-CLUSTER or a FLEXRAY-CLUSTER which is in turn referenced by the NM-CLUSTER</li> </ul> <p>The container name of the NmUserDataPdu container is &lt;PREFIX&gt;&lt;name&gt;_NmComUserData&lt;INSTSUFFIX&gt;.</p> <p>If any PNC-enabled NM-CLUSTER is a FLEXRAY-NM-CLUSTER, an additional <i>ERA</i> (external and internal request array) Pdu container named FrNmPnEiraRxNSdu is created. If any of the PNC-enabled NM-CLUSTER elements is a CAN-NM-CLUSTER, an additional Pdu container named CanNmPnEiraRxNSdu is created. If any of the PNC-enabled NM-CLUSTER elements is a UDP-NM-CLUSTER, an additional Pdu container named UdpNmPnEiraRxNSdu is created.</p> <p>If the following conditions are met, one additional <i>ERA</i> (external request array) Pdu container is created for each COMMUNICATION-CONNECTOR of the ECU-INSTANCE:</p> <ul style="list-style-type: none"> <li>▶ PNC-GATEWAY-TYPE of the COMMUNICATION-CONNECTOR is set to ACTIVE or to PASSIVE.</li> <li>▶ COMMUNICATION-CONNECTOR is connected to an ETHERNET-PHYSICAL-CHANNEL, a CAN-CLUSTER, or a FLEXRAY-CLUSTER, which in turn is related to a PNC-enabled NM-CLUSTER.</li> </ul> <p>The name of the Pdu container is &lt;PREFIX&gt;&lt;channel&gt;NmPnEraRxNSdu, where &lt;channel&gt; is the SHORT-NAME of the FLEXRAY-CLUSTER, CAN-CLUSTER, or ETHERNET-PHYSICAL-CHANNEL.</p> <p>N-PDU elements of LIN-FRAME-TRIGGERING elements which have their IDENTIFIER set to 60 or 61 are not imported.</p>

Configuration parameters	Mapping description
	<p>If a sent N-PDU is referenced by two CAN-TP-CONNECTION elements a second Pdu container is created. The Tx CAN-TP-CONNECTION must reference this N-PDU via DATA-PDU-REF, the Rx CAN-TP-CONNECTION via FLOW-CONTROL-PDU-REF. The direction of the CAN-TP-CONNECTION is determined by the PDU referenced by TP-SDU-REF. The container name of the second Pdu container is &lt;PREFIX&gt;&lt;name&gt;_D&lt;INSTSUFFIX&gt;.</p> <p>PduLength is calculated according to <a href="#">Section 3.3.7, "PDU length calculation"</a>, except for the following cases:</p> <p>For an NmUserDataPdu container that is created for an NM-PDU, PduLength is calculated depending on the type of NM-CLUSTER to which the NM-PDU belongs. First the length of the non-user data is calculated:</p> <ul style="list-style-type: none"> <li>▶ CAN-NM-CLUSTER, UDP-NM-CLUSTER: <math>\max(\text{NM-CBV-POSITION}, \text{NM-NID-POSITION}) + 1</math>. If one of the parameters is not defined or greater than 1, it is assumed to be -1 in the formula above.</li> <li>▶ FLEXRAY-NM-CLUSTER: 2, if NM-NODE-ID-ENABLED of the NM-ECU to which the NM-PDU belongs is set to true, 1 otherwise.</li> </ul> <p><math>\text{PduLength} = (\text{the length of the PDU according to } \text{Section 3.3.7, "PDU length calculation"}) - \langle \text{length of the non-user data} \rangle</math>. If the calculated PduLength is zero or less, a warning is issued and PduLength is not set.</p> <p>For Pdu containers created for PNC-enabled NM-CLUSTER elements PduLength is set to PNC-VECTOR-LENGTH of the SYSTEM. If PNC-VECTOR-LENGTH is not defined a warning is issued and PduLength is not set.</p> <p>SysTPduToFrameMappingRef is set to the AUTOSAR path of the PDU-TO-FRAME-MAPPING that references the PDU. If no PDU-TO-FRAME-MAPPING references the PDU, SysTPduToFrameMappingRef is not configured.</p>

If the configured ECU-INSTANCE contains a server component which receives its requests and sends its responses via Ethernet, the request/response PDUs are extended with Meta-Data that allows the server to keep track of the SocketConnection via which a request was received in order to use the same SocketConnection for the transmission of the response.

Additional Meta-Data is configured if at least one PDU container is created for a PDU instance that fulfills the following conditions:

- ▶ The PDU is of type I-SIGNAL-I-PDU
- ▶ The PDU-TRIGGERING that references the PDU is contained in an ETHERNET-PHYSICAL-CHANNEL

- ▶ The PDU-TRIGGERING is sent or received via a *local* SOCKET-ADDRESS of the configured ECU-INSTANCE on that ETHERNET-PHYSICAL-CHANNEL. For more information about SOCKET-ADDRESS, see [Section 3.4.43, “SoAd”](#).
- ▶ The I-SIGNAL-I-PDU contains exactly one I-SIGNAL
- ▶ The SYSTEM-SIGNAL of the I-SIGNAL is referenced by a CLIENT-SERVER-TO-SIGNAL-MAPPING via CALL-SIGNAL-REF if the configured ECU-INSTANCE receives the I-SIGNAL, or via RETURN-SIGNAL-REF if the configured ECU-INSTANCE sends the I-SIGNAL

In `EcucPduCollection/MetaDataType`, one container is created using `MetaDataTypeSocketConnectionId` as name. Into that container a `MetaDataTypeItem` sub container named `MetaDataTypeItemSocketConnectionId` is inserted. The `MetaDataTypeItemLength` parameter of the sub container is set to 2, the `MetaDataTypeItemType` parameter of the sub container is set to `SOCKET_CONNECTION_ID_16`.

The `MetaDataTypeRef` parameter of every `EcucPduCollection/Pdu` container for which the conditions above apply is configured to refer to the `MetaDataTypeSocketConnectionId` container.

If the configured ECU-INSTANCE sends or receives PDUs via a J-1939-CLUSTER, the PDUs are extended with Meta-Data that allow the J1939 BSW modules to access the CAN-Id that was used for sending or receiving that PDU.

CAN-Id related Meta-Data is configured if at least one PDU container is created for a PDU instance that fulfills the following conditions:

- ▶ The PDU-TRIGGERING that references the PDU is contained in a CAN-PHYSICAL-CHANNEL which in turn is contained in a J-1939-CLUSTER
- ▶ There is a CAN-FRAME which contains a PDU-TO-FRAME-MAPPING that refers to the PDU.
- ▶ There is a CAN-FRAME-TRIGGERING that references the CAN-FRAME and which belongs to the same CAN-PHYSICAL-CHANNEL as the PDU-TRIGGERING.
- ▶ Both CAN-FRAME and PDU are either sent or received by the configured ECU-INSTANCE

In `EcucPduCollection/MetaDataType`, one container is created using `MetaDataTypeCanId` as name. Into that container a `MetaDataTypeItem` sub container named `MetaDataTypeItemCanId` is inserted. The `MetaDataTypeItemLength` parameter of the sub container is set to 4, the `MetaDataTypeItemType` parameter of the sub container is set to `CAN_ID_32`.

The `MetaDataTypeRef` parameter of every `EcucPduCollection/Pdu` container for which the conditions above apply is configured to refer to the `MetaDataTypeCanId` container.

Meta-Data for the J1939Tp module is configured if at least one PDU container has been created for a PDU for which the following conditions are true:

- ▶ The PDU-TRIGGERING that references the PDU is contained in a CAN-PHYSICAL-CHANNE which in turn is contained in a J-1939-CLUSTER

- ▶ There is a J-1939-TP-PG which refers to the PDU.
- ▶ The J-1939-TP-PG is contained in a J-1939-TP-CONNECTION which the configured ECU-INSTANCE either sends or receives.

In `EcucPduCollection/MetaDataType`, one container is created, named `MetaDataTypeJ1939TpSdu`. The following `MetaDataTypeItem` sub containers are added:

ShortName	MetaDataTypeItemLength	MetaDataTypeItemType
Priority	1	PRIORITY_8
SourceAddress	2	SOURCE_ADDRESS_16
TargetAddress	2	TARGET_ADDRESS_16

The `MetaDataTypeRef` parameter of every PDU container that has been set up for a PDU that fulfills the conditions above is configured to refer to the `MetaDataTypeJ1939TpSdu` container.

### 3.4.15. EcuM

Configuration parameters	Mapping description
<code>EcucMConfiguration/EcuMCommonConfiguration/EcuMWakeupSource</code>	<p>For every <code>PNC-MAPPING</code> for which a <code>ComMPnc</code> container has been created in the <code>ComM</code> configuration, see <a href="#">Section 3.4.9, “ComM”</a>, one <code>EcucMWakeupSource</code> container is created if at least one of the following conditions is met:</p> <ul style="list-style-type: none"> <li>▶ The <code>PNC-SYNCHRONOUS-WAKEUP</code> field of the imported <code>ECU-INSTANCE</code> is set to <code>true</code>.</li> <li>▶ The <code>PNC-WAKEUP-ENABLE</code> field of the <code>PNC-MAPPING</code> is set to <code>true</code>.</li> </ul> <p>The name of the <code>EcucMWakeupSource</code> container is <code>ECUM_WKSOURCE_&lt;pncName&gt;</code>, where <code>&lt;pncName&gt;</code> is the name of the <code>ComMPnc</code> container in the <code>ComM</code> module configuration. The <code>EcucMComMPncRef</code> reference is configured to reference that <code>ComMPnc</code> container.</p> <p>For every <code>ETHERNET-PHYSICAL-CHANNEL</code> and for every <code>COMMUNICATION-CLUSTER</code> for which a <code>ComMChannel</code> container has been created in the <code>ComM</code> configuration, see <a href="#">Section 3.4.9, “ComM”</a>, one <code>EcucMWakeupSource</code> container is created if the <code>COMMUNICATION-CONNECTOR</code> that links the <code>ETHERNET-PHYSICAL-CHANNEL</code> or <code>COMMUNICATION-CLUSTER</code> to the imported <code>ECU-INSTANCE</code> has its <code>CREATE-ECU-WAKEUP-SOURCE</code> field set to <code>true</code>. The name of the <code>EcucMWakeupSource</code> container is <code>ECUM_WKSOURCE_&lt;ChannelName&gt;</code>, where <code>&lt;ChannelName&gt;</code> is the name of the <code>ComMChannel</code> container in the <code>ComM</code> module configuration. The</p>



Configuration parameters	Mapping description
	EcuMComMChannelRef reference is configured to reference that ComM-Channel container.

### 3.4.16. Eth

Configuration parameters	Mapping description
EthConfigSet/EthCtrlConfig	<p>Every ETHERNET-COMMUNICATION-CONTROLLER which belongs to the imported ECU-INSTANCE and which is connected to at least one ETHERNET-CLUSTER is considered a <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER of the ECU-INSTANCE. For every <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER of the ECU-INSTANCE, one EthCtrlConfig container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the ETHERNET-COMMUNICATION-CONTROLLER.</p> <p>If the Eth module provides the parameter EthCtrlRxBufLenByte in EthCtrlConfig, this parameter is set to MAXIMUM-RECEIVE-BUFFER-LENGTH. If EthCtrlRxBufLenByte is not present, MAXIMUM-RECEIVE-BUFFER-LENGTH is used to configure EthCtrlConfigIngress/EthCtrlConfigIngressFifo/EthCtrlConfigIngressFifoBufLenByte.</p> <p>If the Eth module provides the parameter EthCtrlTxBufLenByte in EthCtrlConfig, this parameter is set to MAXIMUM-TRANSMIT-BUFFER-LENGTH. If EthCtrlTxBufLenByte is not present, MAXIMUM-TRANSMIT-BUFFER-LENGTH is used to configure EthCtrlConfigEgress/EthCtrlConfigEgressFifo/EthCtrlConfigEgressFifoBufLenByte.</p> <p>EthCtrlPhyAddress is set to MAC-UNICAST-ADDRESS. If MAC-UNICAST-ADDRESS does not represent a valid MAC address, a warning is reported and EthCtrlPhyAddress is not set.</p>

### 3.4.17. EthIf

Configuration parameters	Mapping description
EthIfGeneral	If the imported ECU-INSTANCE has its ETH-SWITCH-PORT-GROUP-DERIVATION set to true and at least one of the ETHERNET-CLUSTER elements to which the imported ECU-INSTANCE is connected has its COUPLING-PORT-SWITCHOFF-DELAY defined, the maximum COUPLING-PORT-SWITCHOFF-

Configuration parameters	Mapping description
	<p>DELAY value of all connected ETHERNET-CLUSTER elements is configured in <code>EthIfSwitchOffPortTimeDelay</code>.</p> <p>If the imported ECU-INSTANCE has its <code>ETH-SWITCH-PORT-GROUP-DERIVATION</code> set to <code>true</code> and at least one of the ETHERNET-CLUSTER elements to which the imported ECU-INSTANCE is connected has its <code>COUPLING-PORT-STARTUP-ACTIVE-TIME</code> defined, the maximum <code>COUPLING-PORT-STARTUP-ACTIVE-TIME</code> value of all connected ETHERNET-CLUSTER elements is configured in <code>EthIfPortStartupActiveTime</code>.</p>
<code>EthIfConfigSet/EthIfPhysController</code>	<p>For each <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE (see <a href="#">Section 3.4.16, “Eth”</a>), an <code>EthIfPhysController</code> container is created. The container name is the <code>SHORT-NAME</code> of the ETHERNET-COMMUNICATION-CONTROLLER.</p> <p><code>EthIfEthCtrlRef</code> references the container created for the <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER in the <code>Eth</code> module.</p>
<code>EthIfConfigSet/EthIfController</code>	<p>A <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE (see <a href="#">Section 3.4.16, “Eth”</a>) is connected to one or more ETHERNET-PHYSICAL-CHANNEL elements. Such a connection exists if an ETHERNET-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE references the ETHERNET-COMMUNICATION-CONTROLLER and is referenced by the ETHERNET-PHYSICAL-CHANNEL. Each of the connections represents a <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER of the ECU-INSTANCE. For each <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE, an <code>EthIfController</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name_cc&gt;_&lt;name_chn&gt;</code>, where <code>&lt;name_cc&gt;</code> is the <code>SHORT-NAME</code> of the ETHERNET-COMMUNICATION-CONTROLLER and <code>&lt;name_chn&gt;</code> is the <code>SHORT-NAME</code> of the ETHERNET-PHYSICAL-CHANNEL.</p> <p><code>EthIfPhysControllerRef</code> references the <code>EthIfPhysController</code> container that has been created for the <i>physical</i> ETHERNET-COMMUNICATION-CONTROLLER.</p> <p>If the ETHERNET-PHYSICAL-CHANNEL that belongs to the virtual ETHERNET-COMMUNICATION-CONTROLLER contains a VLAN that provides a VLAN-IDENTIFIER, <code>EthIfVlanId</code> is set to the value of <code>VLAN-IDENTIFIER</code>.</p> <p>If the ETHERNET-COMMUNICATION-CONNECTOR or the ETHERNET-COMMUNICATION-CONTROLLER provide a valid <code>MAXIMUM-TRANSMISSION-UNIT</code>, <code>EthIfCtrlMtu</code> is set to the value of <code>MAXIMUM-TRANSMISSION-UNIT</code>.</p>

Configuration parameters	Mapping description
	<p>If an <code>EthIfSwitchPortGroup</code> container is created for the <code>ETHERNET-PHYSICAL-CHANNEL</code> of this <code>EthIfController</code> container, <code>EthIfSwitchRefORPortGroupRef</code> is set up to reference that <code>EthIfSwitchPortGroup</code> container.</p>
<code>EthIfConfigSet/EthIfSwitch</code>	<p>For each <code>COUPLING-ELEMENT</code> that has its <code>COUPLING-TYPE</code> set to <code>SWITCH</code> and which references the imported <code>ECU-INSTANCE</code> in <code>ECU-INSTANCE-REF</code>, one <code>EthIfSwitch</code> container is created. The container name is the <code>SHORT-NAME</code> of the <code>COUPLING-ELEMENT</code>.</p> <p><code>EthIfSwitchRef</code> references the container created for the <code>COUPLING-ELEMENT</code> in the <code>EthSwt</code> module.</p>
<code>EthIfConfigSet/EthIfSwitchPortGroup</code>	<p>If the imported <code>ECU-INSTANCE</code> has its <code>ETH-SWITCH-PORT-GROUP-DERIVATION</code> parameter set to <code>true</code>, the configuration algorithm retrieves the list of all <code>COUPLING-PORT</code> elements of any <code>COUPLING-ELEMENT</code> for which an <code>EthIfSwitch</code> container is created.</p> <p>For each <code>ETHERNET-PHYSICAL-CHANNEL</code> and for each <code>PNC-MAPPING</code> that is referenced by any <code>COUPLING-PORT</code>, one <code>EthIfSwitchPortGroup</code> container is created. The container name is either <code>VLANSwtPortGrp_&lt;VLAN&gt;</code>, where <code>&lt;VLAN&gt;</code> is the name of the <code>ETHERNET-PHYSICAL-CHANNEL</code>, or <code>PNCswtPortGrp_&lt;PncID&gt;</code>, where <code>&lt;PncID&gt;</code> is the <code>PNC-IDENTIFIER</code> of the <code>PNC-MAPPING</code>.</p> <p><code>EthIfPortRef</code> references are configured to refer to each <code>EthSwtPort</code> container in <code>EthSwt</code> that corresponds to a <code>COUPLING-PORT</code> which references the <code>ETHERNET-PHYSICAL-CHANNEL</code> or the <code>PNC-MAPPING</code> for which this <code>EthIfSwitchPortGroup</code> container is created.</p> <p><code>EthIfSwitchPortGroupRefSemantics</code> is only configured in <code>EthIfSwitchPortGroup</code> containers that are created for an <code>ETHERNET-PHYSICAL-CHANNEL</code>. If there is any <code>PNC-MAPPING</code> of the configured <code>ECU-INSTANCE</code> which is related to any <code>PDU</code> which is sent over the <code>ETHERNET-PHYSICAL-CHANNEL</code>, the parameter value is set to <code>ETHIF_SWITCH_PORT_GROUP_LINK_INFO</code>, otherwise it is set to <code>ETHIF_SWITCH_PORT_GROUP_CONTROL</code>.</p>

### 3.4.18. EthSM

Configuration parameters	Mapping description
EthSMNetwork	<p>For every ETHERNET-PHYSICAL-CHANNEL that is connected to at least one <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE (see <a href="#">Section 3.4.17, “EthIf”</a>), an EthSMNetwork container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the ETHERNET-PHYSICAL-CHANNEL.</p> <p>EthSMComMNetworkHandleRef references the ComMChannel container in the ComM module configuration, which has been created for the ETHERNET-PHYSICAL-CHANNEL.</p> <p>EthSMEthIfControllerRef references the EthIfController container in the EthIf module configuration, which has been created for the first virtual ETHERNET-COMMUNICATION-CONTROLLER that is connected to the ETHERNET-PHYSICAL-CHANNEL.</p>

### 3.4.19. EthSwt

Configuration parameters	Mapping description
EthSwtConfig	<p>One EthSwtConfig container is created for each COUPLING-ELEMENT that references the imported ECU-INSTANCE via ECU-INSTANCE-REF and has its COUPLING-TYPE set to SWITCH. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COUPLING-ELEMENT.</p>
EthSwtConfig/EthSwtPort	<p>For each COUPLING-PORT contained in a configured COUPLING-ELEMENT one EthSwtPort subcontainer is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COUPLING-PORT.</p> <p>EthSwtPortRole is set depending on the value of COUPLING-PORT/COUPLING-PORT-ROLE:</p> <ul style="list-style-type: none"> <li>▶ ETHSWT_HOST_PORT for HOST-PORT</li> <li>▶ ETHSWT_UP_LINK_PORT for UP-LINK-PORT</li> </ul> <p>For STANDARD-PORT, the parameter is not configured.</p> <p>EthSwtPortMacLayerType is set depending on the value of COUPLING-PORT/MAC-LAYER-TYPE:</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ ETHSWT_PORT_MAC_LAYER_TYPE_XGMII for XG-MII</li> <li>▶ ETHSWT_PORT_MAC_LAYER_TYPE_XMII for X-MII</li> <li>▶ ETHSWT_PORT_MAC_LAYER_TYPE_XXGMII for XXG-MII</li> </ul> <p>EthSwtPortPhysicalLayerType is set depending on the value of COUPLING-PORT/PHYSICAL-LAYER-TYPE:</p> <ul style="list-style-type: none"> <li>▶ ETHSWT_PORT_100BASE_TX for 100BASE-TX</li> <li>▶ ETHSWT_PORT_1000BASE_T for 1000BASE-T</li> <li>▶ ETHSWT_PORT_100BASE_T1 for 100BASE-T1</li> <li>▶ ETHSWT_PORT_1000BASE_T1 for 1000BASE-T1</li> </ul>
EthSwtConfig/EthSwtPort/EthSwtPortPredefinedMacAddresses	<p>The MAC addresses to configure are taken from the MAC-MULTICAST-ADDRESS value of the MAC-MULTICAST-GROUP elements which the COUPLING-PORT references via MAC-MULTICAST-ADDRESS-REFS/MAC-MULTICAST-ADDRESS-REF. For each distinct MAC address value one entry is added to EthSwtPortPredefinedMacAddresses.</p>
EthSwtConfig/EthSwtPort/EthSwtPortIngress	<p>For each COUPLING-PORT, one EthSwtPort/EthSwtPortIngress subcontainer is created.</p> <p>EthSwtPortIngressDefaultVlan is set to VLAN/VLAN-IDENTIFIER if COUPLING-PORT/DEFAULT-VLAN-REF references an ETHERNET-PHYSICAL-CHANNEL that contains a VLAN which in turn defines a valid VLAN-IDENTIFIER.</p> <p>EthSwtPortIngressDefaultPriority is set to VLAN-MEMBERSHIP/DEFAULT-PRIORITY if the VLAN-MEMBERSHIP is contained in COUPLING-PORT/VLAN-MEMBERSHIPS and VLAN-MEMBERSHIP/VLAN-REF references the ETHERNET-PHYSICAL-CHANNEL which is also referenced by COUPLING-PORT/DEFAULT-VLAN-REF. Otherwise, if the DEFAULT-PRIORITY values of all VLAN-MEMBERSHIPS of a COUPLING-PORT are identical, the EthSwtPortIngressDefaultPriority is set to this value.</p> <p>EthSwtPortTrafficClassAssignment is set to COUPLING-PORT-TRAFFIC-CLASS-ASSIGNMENT/TRAFFIC-CLASS if the COUPLING-PORT contains exactly one valid COUPLING-PORT-DETAILS/ETHERNET-TRAFFIC-CLASS-ASSIGNMENTS/ETHERNET-TRAFFIC-CLASS-ASSIGNMENT subelement which in turn contains a valid TRAFFIC-CLASS element but no PRIORITYS/PRIORITY subelements.</p>

Configuration parameters	Mapping description
	<p><b>EthSwtPortTrafficClassAssignment</b> is set to COUPLING-PORT-TRAFFIC-CLASS-ASSIGNMENT/TRAFFIC-CLASS if the COUPLING-PORT contains exactly one valid COUPLING-PORT-DETAILS/ETHERNET-TRAFFIC-CLASS-ASSIGNMENTS/ETHERNET-TRAFFIC-CLASS-ASSIGNMENT subelement which in turn contains a valid TRAFFIC-CLASS element but no PRIORITYS/PRIORITY subelements.</p>
EthSwtConfig/EthSwtPort/EthSwtPortIngress/EthSwtPriorityRegeneration	<p>If the COUPLING-PORT contains a valid COUPLING-PORT-DETAILS subelement, then one EthSwtPriorityRegeneration container is created for each ETHERNET-PRIORITY-REGENERATIONS/ETHERNET-PRIORITY-REGENERATION element contained in COUPLING-PORT/COUPLING-PORT-DETAILS that contains a valid and distinct INGRESS-PRIORITY value. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the ETHERNET-PRIORITY-REGENERATION.</p> <p>EthSwtPriorityRegenerationIngressPriority is set to ETHERNET-PRIORITY-REGENERATION/INGRESS-PRIORITY, EthSwtPriorityRegenerationRegeneratedPriority is set to ETHERNET-PRIORITY-REGENERATION/REGENERATED-PRIORITY.</p>
EthSwtConfig/EthSwtPort/EthSwtPortIngress/EthSwtPriorityTrafficClassAssignment	<p>If the COUPLING-PORT contains a valid COUPLING-PORT-DETAILS subelement and COUPLING-PORT-DETAILS/ETHERNET-TRAFFIC-CLASS-ASSIGNMENTS contains at least one ETHERNET-TRAFFIC-CLASS-ASSIGNMENT element which in turn contains at least one PRIORITY element in ETHERNET-TRAFFIC-CLASS-ASSIGNMENT/PRIORITYS, then one EthSwtPriorityTrafficClassAssignment container is created for each distinct value in PRIORITY. The container name is &lt;PREFIX&gt;&lt;name&gt;_&lt;prio&gt;, where &lt;name&gt; is the SHORT-NAME of the ETHERNET-TRAFFIC-CLASS-ASSIGNMENT and &lt;prio&gt; is the value of ETHERNET-TRAFFIC-CLASS-ASSIGNMENT/PRIORITYS/PRIORITY.</p> <p>EthSwtPriorityTrafficClassAssignmentPriority is set to ETHERNET-TRAFFIC-CLASS-ASSIGNMENT/PRIORITYS/PRIORITY, EthSwtPriorityTrafficClassAssignmentTrafficClass is set to ETHERNET-TRAFFIC-CLASS-ASSIGNMENT/TRAFFIC-CLASS.</p>
EthSwtConfig/EthSwtPort/EthSwtPortEgress	<p>For each COUPLING-PORT that contains at least one entry in COUPLING-PORT-DETAILS/COUPLING-PORT-STRUCTURAL-ELEMENTS, one EthSwtPortEgress container is created.</p> <p>If COUPLING-PORT-DETAILS/COUPLING-PORT-STRUCTURAL-ELEMENTS/LAST-EGRESS-SCHEDULER-REF references a valid COUPLING-PORT-SCHEDULER of the COUPLING-PORT, then EthSwtPortEgressLastSched-</p>

Configuration parameters	Mapping description
	<p>ulerRef is configured to reference the container that has been created for that COUPLING-PORT-SCHEDULER.</p>
EthSwtCon- fig/EthSwt- Port/EthSwt- PortEgress/EthSwt- PortFifo	<p>For each COUPLING-PORT-FIFO subelement of COUPLING-PORT-STRUCTURAL-ELEMENTS, one EthSwtPortFifo container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COUPLING-PORT-FIFO. For each distinct and valid value in COUPLING-PORT-FIFO/ASSIGNED-TRAFFIC-CLASS/ASSIGNED-TRAFFIC-CLASS, one EthSwtPortFifoTrafficClassAssignment subcontainer is created that contains that value.</p>
EthSwtCon- fig/EthSwt- Port/EthSwt- PortEgress/EthSwt- PortShaper	<p>For each COUPLING-PORT-SHAPER subelement of COUPLING-PORT-STRUCTURAL-ELEMENTS, one EthSwtPortShaper container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COUPLING-PORT-SHAPER.</p> <p>EthSwtPortShaperIdleSlope is set to COUPLING-PORT-SHAPER/IDLE-SLOPE, EthSwtPortEgressPredecessorFifoRef is configured to reference the EthSwtPortFifo container that has been created for the COUPLING-PORT-FIFO which COUPLING-PORT-SHAPER/REDECESSOR-FIFO-REF references.</p>
EthSwtCon- fig/EthSwt- Port/EthSwt- PortEgress/EthSwt- PortScheduler	<p>For each COUPLING-PORT-SCHEDULER subelement of COUPLING-PORT-STRUCTURAL-ELEMENTS, one EthSwtPortScheduler container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the COUPLING-PORT-SCHEDULER.</p> <p>EthSwtPortSchedulerAlgorithm is set depending on the value of COUPLING-PORT-SCHEDULER/PORT-SCHEDULER:</p> <ul style="list-style-type: none"> <li>▶ ETHSWT_SCHEDULER_DEFICIT_ROUND_ROBIN for DEFICIT-ROUND-ROBIN</li> <li>▶ ETHSWT_SCHEDULER_STRICT_PRIORITY for STRICT-PRIORITY</li> <li>▶ ETHSWT_SCHEDULER_WEIGHTED_ROUND_ROBIN for WEIGHTED-ROUND-ROBIN</li> </ul> <p>For each COUPLING-PORT-STRUCTURAL-ELEMENT referenced by COUPLING-PORT-SCHEDULER/PREDECESSOR-REFS/PREDECESSOR-REF one EthSwtPortSchedulerPredecessor subcontainer is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the referenced COUPLING-PORT-STRUCTURAL-ELEMENT.</p>

Configuration parameters	Mapping description
	<p>EthSwtPortSchedulerPredecessorOrder is set to <math>10 * (&lt;numberOfElementsInPredecessorList&gt; - &lt;positionInPredecessorList&gt;)</math>, i.e. 0 for the last referenced element in COUPLING-PORT-SCHEDULER/PREDECESSOR-REFS, 10 for the second to last referenced, 20 for the third to last referenced element, etc. EthSwtPortEgressPredecessorRef is configured to reference the container that has been created for the COUPLING-PORT-STRUCTURAL-ELEMENT which COUPLING-PORT-SCHEDULER/PREDECESSOR-REFS/PREDECESSOR-REF references.</p>
EthSwtConfig/EthSwtPort/EthSwtPortVlanMembership	<p>The VLAN ID of a VLAN-MEMBERSHIP is the VLAN-IDENTIFIER of the VLAN contained in the ETHERNET-PHYSICAL-CHANNEL that VLAN-MEMBERSHIP/VLAN-REF refers to. For each VLAN-MEMBERSHIP that is associated with a distinct valid VLAN ID, one EthSwtPortVlanMembership container is created. The container name is EthSwtPortVlanMembership_&lt;VLAN_ID&gt;, where &lt;VLAN_ID&gt; is the associated VLAN ID.</p> <p>EthSwtPortVlanMembershipId is set to the VLAN ID.</p> <p>EthSwtPortVlanForwardingType is set depending on the value of VLAN-MEMBERSHIP/SEND-ACTIVITY:</p> <ul style="list-style-type: none"> <li>▶ ETHSWT_NOT_SENT for NOT-SENT</li> <li>▶ ETHSWT_SENT_TAGGED for SENT-TAGGED</li> <li>▶ ETHSWT_SENT_UNTAGGED for SENT-UNTAGGED</li> </ul>

### 3.4.20. EthTSyn

Configuration parameters	Mapping description
EthTSynGeneral	<p>One EthTSynGeneral container is created if there is at least one GLOBAL-TIME-DOMAIN that contains a GLOBAL-TIME-ETH-MASTER or a GLOBAL-TIME-ETH-SLAVE element that references an ETHERNET-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE.</p> <p>EthTSynMessageCompliance is configured if all of the GLOBAL-TIME-DOMAIN elements that contain a ETH-GLOBAL-TIME-DOMAIN-PROPS element providing a valid value in MESSAGE-COMPLIANCE have that value set consistently. If that value is IEEE802-1AS, EthTSynMessageCompliance is set to true. If that value is IEEE802-1AS-AUTOSAR, EthTSynMessageCompliance is set to false.</p>



Configuration parameters	Mapping description
	<p>The parameter <code>EthTSynDestPhyAddr</code> is configured by collecting all valid <code>DESTINATION-PHYSICAL-ADDRESS</code> values of <code>ETH-GLOBAL-TIME-DOMAIN-PROPS</code> elements that are aggregated by <code>GLOBAL-TIME-DOMAIN</code> elements of the imported <code>ECU-INSTANCE</code>. If the collected values refer to the same valid MAC address, that address is used to configure <code>EthTSynDestPhyAddr</code>.</p>
<code>EthTSynGlobalTimeDomain</code>	<p>One <code>EthTSynGlobalTimeDomain</code> container is created for each <code>GLOBAL-TIME-DOMAIN</code> that contains a <code>GLOBAL-TIME-ETH-MASTER</code> or a <code>GLOBAL-TIME-ETH-SLAVE</code> element that references an <code>ETHERNET-COMMUNICATION-CONNECTOR</code> of the imported <code>ECU-INSTANCE</code>. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>GLOBAL-TIME-DOMAIN</code>.</p> <p><code>EthTSynGlobalTimeDomainId</code> is set to <code>DOMAIN-ID</code>.</p> <p><code>EthTSynSynchronizedTimeBaseRef</code> references the corresponding <code>StbMSynchronizedTimeBase</code> container in the <code>StbM</code> module. If the <code>GLOBAL-TIME-DOMAIN</code> is a subdomain of a parent <code>GLOBAL-TIME-DOMAIN</code>, <code>EthTSynSynchronizedTimeBaseRef</code> references the <code>StbMSynchronizedTimeBase</code> container that has been created for the parent <code>GLOBAL-TIME-DOMAIN</code>. For further information about the configuration of the <code>StbM</code> module, see <a href="#">Section 3.4.45, “StbM”</a></p> <p>If the release version of <code>EthTSyn</code> is 4.4.0 or newer, <code>EthTSynGlobalTimeEthIfRef</code> references the <code>EthIfController</code> container that has been created for the <code>ETHERNET-COMMUNICATION-CONNECTOR</code> to which the <code>GLOBAL-TIME-ETH-MASTER</code> or <code>GLOBAL-TIME-ETH-SLAVE</code> refers. For <code>EthTSyn</code> versions older than 4.4.0 the parameter is not configured. For further information about the configuration of the <code>EthIf</code> module, see <a href="#">Section 3.4.17, “EthIf”</a>.</p> <p>If the release version of <code>EthTSyn</code> is 4.4.0 or newer and the Ethernet <code>GLOBAL-TIME-DOMAIN</code> contains a valid <code>ETH-GLOBAL-TIME-DOMAIN-PROPS</code> subelement that in turn contains a valid <code>VLAN-PRIORITY</code> subelement, <code>EthTSynFramePrio</code> is configured to hold the value of <code>VLAN-PRIORITY</code>. For <code>EthTSyn</code> versions older than 4.4.0 the parameter is not configured.</p> <p>If the release version of <code>EthTSyn</code> is 4.4.0 or newer, <code>EthTSynGlobalTimeDebounceTime</code> is set to <code>GLOBAL-TIME-DOMAIN/DEBOUNCE-TIME</code>. For <code>EthTSyn</code> versions older than 4.4.0 the parameter is not configured.</p>
<code>EthTSynGlobalTimeDomain/EthTSyn-</code>	<p>If the Ethernet <code>GLOBAL-TIME-DOMAIN</code> contains a valid <code>ETH-GLOBAL-TIME-DOMAIN-PROPS</code> subelement that contains a non-empty ID list in <code>FUP-DATA-ID-LISTS</code>, that list is used to configure the container list <code>EthTSynGlobal-</code></p>

Configuration parameters	Mapping description
GlobalTimeFollowUpDataIDList	<p>TimeDomain/EthTSynGlobalTimeFollowUpDataIDList/EthTSynGlobalTimeFollowUpDataIDListElement.</p> <p>One subcontainer is created per element named <code>Element_&lt;idx&gt;</code>, where <code>&lt;idx&gt;</code> is the zero-based index of the element within the list. <code>EthTSynGlobalTimeFollowUpDataIDListIndex</code> contains the zero-based index of the element within the list, <code>EthTSynGlobalTimeFollowUpDataIDListValue</code> is configured to contain the value of the element.</p>
EthTSynGlobalTimeDomain/EthTSynPortConfig	<p>One <code>EthTSynGlobalTimeDomain</code> is created for the <code>ETHERNET-COMMUNICATION-CONNECTOR</code> which connects the imported <code>ECU-INSTANCE</code> to the Ethernet <code>GLOBAL-TIME-DOMAIN</code>. The container name is the same as the name of the parent <code>EthTSynGlobalTimeDomain</code> container.</p> <p>If the <code>GLOBAL-TIME-DOMAIN</code> for which the <code>EthTSynGlobalTimeDomain</code> container has been created contains an <code>ETH-GLOBAL-TIME-DOMAIN-PROPS</code> element that in turn contains one or more <code>ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT</code> elements that refer to a <code>COUPLING-PORT</code> for which a <code>EthSwtPort</code> container has been created, see <a href="#">Section 3.4.19, “EthSwt”</a>, then one additional <code>EthTSynPortConfig</code> is created for each of these <code>ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT</code> elements. In that case, the container name is the <code>SHORT-NAME</code> of the <code>COUPLING-PORT</code> and <code>EthTSynSwitchManagementEthSwitchPortRef</code> is configured to refer to the <code>EthSwtPort</code> container in the <code>EthSwt</code> module.</p> <p>If the release version of <code>EthTSyn</code> is older than 4.4.0, <code>EthTSynGlobalTimeEthIfRef</code> references the <code>EthIfController</code> container that has been created for the <code>ETHERNET-COMMUNICATION-CONNECTOR</code> to which the <code>GLOBAL-TIME-ETH-MASTER</code> or <code>GLOBAL-TIME-ETH-SLAVE</code> refers. For <code>EthTSyn</code> versions 4.4.0 and newer the parameter is not configured. For further information about the configuration of the <code>EthIf</code> module, see <a href="#">Section 3.4.17, “EthIf”</a>.</p> <p>If the release version of <code>EthTSyn</code> is older than 4.4.0 and the Ethernet <code>GLOBAL-TIME-DOMAIN</code> contains a valid <code>ETH-GLOBAL-TIME-DOMAIN-PROPS</code> subelement that in turn contains a valid <code>VLAN-PRIORITY</code> subelement, <code>EthTSynFramePrio</code> is configured to hold the value of <code>VLAN-PRIORITY</code>. For <code>EthTSyn</code> versions 4.4.0 and newer the parameter is not configured.</p> <p>If the release version of <code>EthTSyn</code> is older than 4.4.0, <code>EthTSynGlobalTimeDebounceTime</code> is set to <code>GLOBAL-TIME-DOMAIN/DEBOUNCE-TIME</code>. For <code>EthTSyn</code> versions 4.4.0 and newer the parameter is not configured.</p>

Configuration parameters	Mapping description
EthTSynGlobalTimeDomain/EthTSynPortConfig/EthTSynPdelayConfig	<p>The EthTSynPdelayConfig container is created for each ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT of the Ethernet GLOBAL-TIME-DOMAIN that refers to a COUPLING-PORT that either belongs to an ETHERNET-COMMUNICATION-CONTROLLER or to an COUPLING-ELEMENT of the imported ECU-INSTANCE.</p> <p>EthTSynGlobalTimePdelayRespEnable is set to ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT/PDELAY-RESPONSE-ENABLED.</p> <p>EthTSynGlobalTimePropagationDelay is set to COUPLING-PORT/COUPLING-PORT-DETAILS/GLOBAL-TIME-PROPS/PROPAGATION-DELAY of the COUPLING-PORT that is referenced by ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT/COUPLING-PORT-REF.</p> <p>EthTSynGlobalTimeTxPdelayReqPeriod is set to ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT/PDELAY-REQUEST-PERIOD.</p> <p>EthTSynPdelayLatencyThreshold is set to ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT/PDELAY-LATENCY-THRESHOLD.</p> <p>EthTSynPdelayRespAndRespFollowUpTimeout is set to ETH-GLOBAL-TIME-MANAGED-COUPLING-PORT/PDELAY-RESP-AND-RESP-FOLLOW-UP-TIMEOUT.</p>
EthTSynGlobalTimeMaster	<p>This container is only configured if the parent EthTSynPortConfig container has been created for the ETHERNET-COMMUNICATION-CONNECTOR which connects the imported ECU-INSTANCE to the Ethernet GLOBAL-TIME-DOMAIN and the GLOBAL-TIME-ETH-MASTER of the GLOBAL-TIME-DOMAIN references that ETHERNET-COMMUNICATION-CONNECTOR.</p> <p>If the release version of EthTSyn is 4.4.0 or newer, the EthTSynGlobalTimeMaster container is added as EthTSynGlobalTimeDomain child container. If the release version is older than 4.4.0, the container is added as EthTSynGlobalTimeDomain/EthTSynPortConfig child container.</p> <p>EthTSynGlobalTimeTxPeriod is set to SYNC-PERIOD of the GLOBAL-TIME-ETH-MASTER.</p> <p>EthTSynCyclicMsgResumeTime is set to IMMEDIATE-RESUME-TIME of the GLOBAL-TIME-ETH-MASTER.</p> <p>EthTSynGlobalTimeTxCrcSecured is set to CRC-SECURED of the GLOBAL-TIME-ETH-MASTER.</p>

Configuration parameters	Mapping description
	<p>EthTSynTLVFollowUpOFSSubTLV is set to GLOBAL-TIME-ETH-MASTER/SUB-TLV-CONFIG/OFS-SUB-TLV.</p> <p>EthTSynTLVFollowUpStatusSubTLV is set to GLOBAL-TIME-ETH-MASTER/SUB-TLV-CONFIG/STATUS-SUB-TLV.</p> <p>EthTSynTLVFollowUpTimeSubTLV is set to GLOBAL-TIME-ETH-MASTER/SUB-TLV-CONFIG/TIME-SUB-TLV.</p> <p>EthTSynTLVFollowUpUserDataSubTLV is set to GLOBAL-TIME-ETH-MASTER/SUB-TLV-CONFIG/USER-DATA-SUB-TLV.</p>
EthTSynGlobal-TimeMaster/EthTSyn-CrcTimeFlagsTxSecured	<p>If the GLOBAL-TIME-DOMAIN contains an ETH-GLOBAL-TIME-DO-MAIN-PROPS sub element which in turn contains an CRC-FLAGS element, that sub element is used to configure parameters in the EthTSynCrcTimeFlagsTxSecured sub container. The container name is the same as the container name of the EthTSynGlobalTimeDomain.</p> <p>EthTSynCrcCorrectionField is set to CRC-FLAGS/CRC-CORRECTION-FIELD.</p> <p>EthTSynCrcDomainNumber is set to CRC-FLAGS/CRC-DOMAIN-NUMBER.</p> <p>EthTSynCrcMessageLength is set to CRC-FLAGS/CRC-MESSAGE-LENGTH.</p> <p>EthTSynCrcPreciseOriginTimestamp is set to CRC-FLAGS/CRC-PRECISE-ORIGIN-TIMESTAMP.</p> <p>EthTSynCrcSequenceId is set to CRC-FLAGS/CRC-SEQUENCE-ID.</p> <p>EthTSynCrcSourcePortIdentity is set to CRC-FLAGS/CRC-SOURCE-PORT-IDENTITY.</p>
EthTSynGlobal-TimeSlave	<p>This container is only configured if the parent EthTSynPortConfig container has been created for the ETHERNET-COMMUNICATION-CONNECTOR which connects the imported ECU-INSTANCE to the Ethernet GLOBAL-TIME-DOMAIN and one of the GLOBAL-TIME-ETH-SLAVE elements of the GLOBAL-TIME-DOMAIN references that ETHERNET-COMMUNICATION-CONNECTOR.</p> <p>If the release version of EthTSyn is 4.4.0 or newer, the EthTSynGlobal-TimeSlave container is added as EthTSynGlobalTimeDomain child container. If the release version is older than 4.4.0, the container is added as EthTSynGlobalTimeDomain/EthTSynPortConfig child container.</p>

Configuration parameters	Mapping description
	<p>EthTSynRxCrcValidated is set to CRC-VALIDATED of the GLOBAL-TIME-ETH-SLAVE.</p> <p>EthTSynGlobalTimeFollowUpTimeout is set to FOLLOW-UP-TIME-OUT-VALUE of the GLOBAL-TIME-ETH-SLAVE.</p>
EthTSynGlobal-TimeSlave/EthTSyn-CrcFlagsRxValidated	<p>If the GLOBAL-TIME-DOMAIN contains an ETH-GLOBAL-TIME-DO-MAIN-PROPS sub element which in turn contains an CRC-FLAGS element, that sub element is used to configure parameters in the EthTSynCrcFlagsRxValidated sub container. The container name is the same as the container name of the EthTSynGlobalTimeDomain.</p> <p>The same parameters are configured as described for EthTSynGlobal-TimeMaster/EthTSynCrcTimeFlagsTxSecured.</p>

### 3.4.21. FiM

Configuration parameters	Mapping description
FiM/FiMConfigSet/FiMFID	A FiMFID is created for each DiagnosticFunctionIdentifier.
FiMConfigSet/FiMSummaryEvent	<p>Based on the attributes ActualEvent or AliasEvent of the Diagnostic-FimAliasEventMapping a FiMSummaryEvent container is created for every mapping, as follows:</p> <ol style="list-style-type: none"> <li>1. If the ActualEvent attribute exists then the FiMSummaryEvent is created based on it.</li> <li>2. If the AliasEvent attribute exists then the FiMSummaryEvent is created based on it.</li> </ol> <p>After going through all the DiagnosticFimAliasEventMapping, each Fi-mAliasEventGroupMapping is taken into account, and based on the DiagnosticFimEventGroup and the DiagnosticFimAliasEventGroup attributes a FiMSummaryEvent is created for every mapping as follows:</p> <ol style="list-style-type: none"> <li>1. If the DiagnosticFimEventGroup attribute exists then the FiMSummaryEvent is created based on the actual events that it contains, and which were not covered by the DiagnosticFimAliasEventMapping.</li> <li>2. If the DiagnosticFimAliasEventGroup attribute exists then the FiMSummaryEvent is created based on the alias events that it contains,</li> </ol>

Configuration parameters	Mapping description
	and which were not covered by the DiagnosticFimAliasEventMapping.
FiMConfigSet/FiMinhibitionConfiguration	For every DiagnosticFunctionIdentifierInhibit a container of type FiMinhibitionConfiguration is created. The name of the container is the short name of DiagnosticFunctionIdentifierInhibit.
FiMConfigSet/FiMinhibitionConfiguration/DiagnosticFunctionIdentifierInhibit/FiMinhEventId	For every FiMinhibitionConfiguration container a FiMinhEventId sub-container is created with the short name FiMinhEventId. The FiMinhEventId has a FiMinhRefChoice container of type choice.
FiMConfigSet/FiMEventSummary/FiMInputSumEventRef	For every FiMEventSummary container a mandatory FiMInputSumEventRef reference is set.
FiMConfigSet/FiMEventSummary/FiMOutputSumEventRef	For every FiMEventSummary container a mandatory FiMOutputSumEventRef reference is set.
FiMConfigSet/FiMSummaryEventId	<p>For each FimAliasEventGroupMapping a FiMSummaryEventId container is created based on the attributes DiagnosticFimEventGroup and DiagnosticFimAliasEventGroup as follows:</p> <ol style="list-style-type: none"> <li>1. If the DiagnosticFimEventGroup attribute exists then the FiMSummaryEventId is created based on the actual eventGroup.</li> <li>2. If the DiagnosticFimAliasEventGroup attribute exists then the FiMSummaryEventId is created based on the alias eventGroup.</li> </ol>

### 3.4.22. Fr

Configuration parameters	Mapping description
FrMultipleConfiguration/FrController	For every FLEXRAY-COMMUNICATION-CONTROLLER which is connected to a FLEXRAY-CLUSTER and which belongs to the imported ECU-INSTANCE, a FrController container is created. The container name is <PREFIX><name>, where <name> is the SHORT-NAME of the FLEXRAY-COMMUNICATION-CONTROLLER.

Configuration parameters	Mapping description
	<p><b>FrPAllowHaltDueToClock</b> is set to ALLOW-HALT-DUE-TO-CLOCK.</p> <p><b>FrPAllowPassiveToActive</b> is set to ALLOW-PASSIVE-TO-ACTIVE.</p> <p><b>FrPChannels</b> is set depending on which FLEXRAY-PHYSICAL-CHANNEL elements the FLEXRAY-COMMUNICATION-CONTROLLER is connected to:</p> <ul style="list-style-type: none"> <li>▶ <b>FR_CHANNEL_A</b> if connected to CHANNEL-A.</li> <li>▶ <b>FR_CHANNEL_B</b> if connected to CHANNEL-B.</li> <li>▶ <b>FR_CHANNEL_AB</b> if connected to CHANNEL-A and CHANNEL-B.</li> </ul> <p><b>FrPClusterDriftDamping</b> is set to CLUSTER-DRIFT-DAMPING.</p> <p><b>FrPdAcceptedStartupRange</b> is set to ACCEPTED-STARTUP-RANGE.</p> <p><b>FrPDecodingCorrection</b> is set to DECODING-CORRECTION.</p> <p><b>FrPDelayCompensationA</b> is set to DELAY-COMPENSATION-A.</p> <p><b>FrPDelayCompensationB</b> is set to DELAY-COMPENSATION-B.</p> <p><b>FrPdListenTimeout</b> is set to LISTEN-TIMEOUT.</p> <p><b>FrPdMicrotick</b> is set depending on the value of MICROTICK-DURATION [ns]:</p> <ul style="list-style-type: none"> <li>▶ <b>T200NS</b> for 200.</li> <li>▶ <b>T100NS</b> for 100.</li> <li>▶ <b>T50NS</b> for 50.</li> <li>▶ <b>T25NS</b> for 25.</li> <li>▶ <b>T12_5NS</b> for 12.5.</li> <li>▶ For any other values <b>FrPdMicrotick</b> is not configured and a warning is reported.</li> </ul> <p><b>FrPExternalSync</b> is set to EXTERNAL-SYNC.</p> <p><b>FrPFallBackInternal</b> is set to FALL-BACK-INTERNAL.</p> <p><b>FrPKeySlotId</b> is set to KEY-SLOT-ID. If KEY-SLOT-ONLY-ENABLED, KEY-SLOT-USED-FOR-START-UP, or KEY-SLOT-USED-FOR-SYNC are set to true, but KEY-SLOT-ID is undefined, a warning is reported.</p> <p><b>FrPKeySlotOnlyEnabled</b> is set to KEY-SLOT-ONLY-ENABLED.</p>

Configuration parameters	Mapping description
	<p>FrPKeySlotUsedForStartup is set to KEY-SLOT-USED-FOR-START-UP.</p> <p>FrPKeySlotUsedForSync is set to KEY-SLOT-USED-FOR-SYNC.</p> <p>FrPLatestTx is set to LATEST-TX.</p> <p>FrPMacroInitialOffsetA is set to MACRO-INITIAL-OFFSET-A.</p> <p>FrPMacroInitialOffsetB is set to MACRO-INITIAL-OFFSET-B.</p> <p>FrPMicroInitialOffsetA is set to MICRO-INITIAL-OFFSET-A.</p> <p>FrPMicroInitialOffsetB is set to MICRO-INITIAL-OFFSET-B.</p> <p>FrPMicroPerCycle is set to MICRO-PER-CYCLE.</p> <p>FrPNmVectorEarlyUpdate is set to NM-VECTOR-EARLY-UPDATE.</p> <p>FrPOffsetCorrectionOut is set to OFFSET-CORRECTION-OUT.</p> <p>FrPOffsetCorrectionStart is set to OFFSET-CORRECTION-START of the FLEXRAY-CLUSTER to which the FLEXRAY-COMMUNICATION-CONTROLLER is connected.</p> <p>FrPPayloadLengthDynMax is set to MAXIMUM-DYNAMIC-PAYLOAD-LENGTH.</p> <p>FrPRateCorrectionOut is set to RATE-CORRECTION-OUT.</p> <p>FrPSamplesPerMicrotick is set depending on the value of SAMPLES-PER-MICROTICK:</p> <ul style="list-style-type: none"> <li>▶ N1SAMPLES for 1.</li> <li>▶ N2SAMPLES for 2.</li> <li>▶ N4SAMPLES for 4.</li> <li>▶ For all other values of SAMPLES-PER-MICROTICK FrPSamplesPerMicrotick is set to N1SAMPLES and a warning is reported.</li> </ul> <p>FrPSecondKeySlotId is set to SECOND-KEY-SLOT-ID. If TWO-KEY-SLOT-MODE is set to true, but SECOND-KEY-SLOT-ID is undefined, a warning is reported.</p> <p>FrPTwoKeySlotMode is set to TWO-KEY-SLOT-MODE</p> <p>FrPWakeupChannel is set depending on the COMMUNICATION-CONNECTOR elements of the FLEXRAY-COMMUNICATION-CONTROLLER. The PHYSI-</p>



Configuration parameters	Mapping description
	<p>CAL-CHANNEL which is connected to the COMMUNICATION-CONNECTOR which has WAKE-UP-CHANNEL set to true determines the value of FrPwakeupChannel:</p> <ul style="list-style-type: none"> <li>▶ FR_CHANNEL_A if the CHANNEL-NAME is CHANNEL-A.</li> <li>▶ FR_CHANNEL_B if the CHANNEL-NAME is CHANNEL-B.</li> </ul> <p>If no COMMUNICATION-CONNECTOR has its WAKE-UP-CHANNEL set to true a warning is issued and FrPwakeupChannel is set to FR_CHANNEL_A.</p> <p>FrPwakeupPattern is set to WAKE-UP-PATTERN.</p>
FrMultipleConfiguration/FrController/FrAbsoluteTimer	If no FrAbsoluteTimer container exists yet, a new one named FrAbsoluteTimer is created.

### 3.4.23. FrArTp

Configuration parameters	Mapping description
FrArTpMultipleConfiguration/FrArTpChannel	<p>For each FLEXRAY-AR-TP-CHANNEL that contains at least one FLEXRAY-AR-TP-CONNECTION which is processed by the imported ECU-INSTANCE, one FrArTpChannel container is created. The name of the container is FrArTpChannel_&lt;idx&gt;, where &lt;idx&gt; is a zero-based index.</p> <p>FrArTpAckType is configured depending on the value of ACK-TYPE:</p> <ul style="list-style-type: none"> <li>▶ FRARTP_ACK_WITH_RT for ACK-WITH-RT.</li> <li>▶ FRARTP_ACK_WITHOUT_RT for ACK-WITHOUT-RT.</li> <li>▶ FRARTP_NO for NO-ACK.</li> </ul> <p>FrArTpAdrType is configured depending on the value of EXTENDED-ADDRESSING:</p> <ul style="list-style-type: none"> <li>▶ FRARTP_TB for true.</li> <li>▶ FRARTP_OB for false.</li> </ul> <p>FrArTpGrpSeg is set to MULTICAST-SEGMENTATION.</p>

Configuration parameters	Mapping description
	<p>FrArTpLm is configured depending on the value of MAXIMUM-MESSAGE-LENGTH:</p> <ul style="list-style-type: none"> <li>▶ FRARTP_ISO6 for ISO-6.</li> <li>▶ FRARTP_ISO for ISO.</li> <li>▶ FRARTP_L4G for I4G.</li> </ul> <p>FrArTpMaxAr is set to MAX-AR.</p> <p>FrArTpMaxAs is set to MAX-AS.</p> <p>FrArTpMaxBs is set to MAX-BE.</p> <p>FrArTpMaxRn is set to MAX-RETRIES.</p> <p>FrArTpMaxWft is set to MAX-FC-WAIT.</p> <p>FrArTpStMin is set to MINIMUM-SEPARATION-TIME.</p> <p>FrArTpStMinGrpSeg is set to MINIMUM-MULTICAST-SEPERATION-TIME.</p> <p>FrArTpTc is set to CANCELLATION.</p> <p>FrArTpTimeBr is set to TIME-BR.</p> <p>FrArTpTimeCs is set to TIME-CS.</p> <p>FrArTpTimeoutAr is set to TIMEOUT-AR.</p> <p>FrArTpTimeoutAs is set to TIMEOUT-AS.</p> <p>FrArTpTimeoutBs is set to TIMEOUT-BE.</p> <p>FrArTpTimeoutCr is set to TIMEOUT-CR.</p>
FrTpMultipleConfig/FrArTpChannel/FrArTpConnection	<p>The imported ECU-INSTANCE processes a FLEXRAY-AR-TP-CONNECTION if the FLEXRAY-AR-TP-CONNECTION references via SOURCE-REF or TARGET-REFS/TARGET-REF any FLEXRAY-AR-TP-NODE that belongs to the imported ECU-INSTANCE.</p> <p>A FLEXRAY-AR-TP-NODE belongs to the imported ECU-INSTANCE if it references any COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE via CONNECTOR-REF.</p>

Configuration parameters	Mapping description
	<p>The sender FLEXRAY-AR-TP-NODE of a FLEXRAY-AR-TP-CONNECTION is the FLEXRAY-AR-TP-NODE which the FLEXRAY-AR-TP-CONNECTION references via SOURCE-REF</p> <p>The receiver FLEXRAY-AR-TP-NODE elements of a FLEXRAY-AR-TP-CONNECTION are the FLEXRAY-AR-TP-NODE elements which the FLEXRAY-AR-TP-CONNECTION references via TARGET-REFS/TARGET-REF.</p> <p>The TP-ADDRESS of a FLEXRAY-AR-TP-NODE is the TP-ADDRESS which it references via TP-ADDRESS-REF.</p> <p>The sender TP-ADDRESS of a FLEXRAY-AR-TP-CONNECTION is the TP-ADDRESS of the sender node of the FLEXRAY-AR-TP-CONNECTION. The receiver TP-ADDRESS of a FLEXRAY-AR-TP-CONNECTION is either the TP-ADDRESS which the FLEXRAY-AR-TP-CONNECTION references via MULTICAST-REF or the TP-ADDRESS of the first receiver FLEXRAY-AR-TP-NODE if MULTICAST-REF is not defined.</p> <p>If the sender FLEXRAY-AR-TP-NODE belongs to the imported ECU-INSTANCE, its TP-ADDRESS is considered the <i>local</i> address of the FLEXRAY-AR-TP-CONNECTION, and the receiver TP-ADDRESS is considered as its <i>remote</i> TP-ADDRESS. If the sender FLEXRAY-AR-TP-NODE does not belong to the imported ECU-INSTANCE, its TP-ADDRESS is considered as <i>remote</i> and the receiver TP-ADDRESS is considered as <i>local</i>.</p> <p>All FLEXRAY-AR-TP-CONNECTION elements of a FLEXRAY-AR-TP-CHANNEL that are processed by the imported ECU-INSTANCE are collected according to their <i>local</i> and <i>remote</i> address pair. For each unique <i>local/remote</i> address pair, one FlexrayArTpConnection container is created.</p> <p>FrArTpLa is set to TP-ADDRESS/TP-ADDRESS of the <i>local</i> TP-ADDRESS.</p> <p>FrArTpMultRec is set to true if the FLEXRAY-AR-TP-CONNECTION references a TP-ADDRESS via MULTICAST-REF, otherwise to false.</p> <p>FrArTpRa is set to TP-ADDRESS/TP-ADDRESS of the <i>remote</i> TP-ADDRESS.</p>
FrTpMultipleConfig/FrArTpChannel/FrArTpConnection/FrArTpRxSdu	<p>If the imported ECU-INSTANCE receives any PDU which the FLEXRAY-AR-TP-CONNECTION references either via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF, an FrArTpRxSdu sub container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p>

Configuration parameters	Mapping description
	FrArTpRxSduRef is configured to reference the corresponding container of the PDU in the EcuC module.
FrTpMultipleConfig/FrArTpChannel/FrArTpConnection/FrArTpTxSdu	<p>If the imported ECU-INSTANCE sends any PDU which the FLEXRAY-AR-TP-CONNECTION references either via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF, an FrArTpTxSdu sub container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>FrArTpTxSduRef is configured to reference the corresponding container of the PDU in the EcuC module.</p>
FrTpMultipleConfig/FrArTpChannel/FrArTpPdu	<p>For each N-PDU which the FLEXRAY-AR-TP-CHANNEL references via N-PDU-REFS/N-PDU-REF and which the imported ECU-INSTANCE either receives or sends, one FrArTpPdu container is created. <a href="#">Section 3.3.5, "Collection of N-PDU elements"</a> describes the cases in which an FrArTpPdu is considered as received or sent by the imported ECU-INSTANCE. The name of the container is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the N-PDU.</p> <p>FrArTpPduRef is configured to reference the corresponding container of the N-PDU in the EcuC module.</p> <p>FrArTpPduDirection is set to FRARTP_RX if the imported ECU-INSTANCE receives the N-PDU. If the imported ECU-INSTANCE sends the N-PDU, FrArTpPduDirection is set to FRARTP_TX.</p>

### 3.4.24. FrIf

Configuration parameters	Mapping description
FrIfConfig/FrIfCluster	<p>For every FLEXRAY-CLUSTER to which the imported ECU-INSTANCE is connected a FrIfCluster container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the FLEXRAY-CLUSTER.</p> <p>FrIfGdWakeupRxIdle is set to WAKEUP-RX-IDLE.</p> <p>FrIfGdWakeupRxLow is set to WAKEUP-RX-LOW.</p> <p>FrIfGdWakeupRxWindow is set to WAKEUP-RX-WINDOW.</p>

Configuration parameters	Mapping description
	<p>FrIfGdWakeupTxIdle is set to WAKEUP-TX-IDLE.</p> <p>FrIfGPayloadLengthStatic is set to PAYLOAD-LENGTH-STATIC.</p> <p>FrIfGNetworkManagementVectorLength is set to NETWORK-MANAGEMENT-VECTOR-LENGTH.</p> <p>FrIfGdCycle is set to CYCLE.</p> <p>FrIfGdCasRxLowMax is set to CAS-RX-LOW-MAX.</p> <p>FrIfGNumberOfStaticSlots is set to NUMBER-OF-STATIC-SLOTS.</p> <p>FrIfGColdStartAttempts is set to COLD-START-ATTEMPTS.</p> <p>FrIfGdActionPointOffset is set to ACTION-POINT-OFFSET.</p> <p>FrIfGdDynamicSlotIdlePhase is set to DYNAMIC-SLOT-IDLE-PHASE.</p> <p>FrIfGdMinislot is set to MINISLOT-DURATION.</p> <p>FrIfGdMiniSlotActionPointOffset is set to MINISLOT-ACTION-POINT-OFFSET.</p> <p>FrIfGdNit is set to NETWORK-IDLE-TIME.</p> <p>FrIfGdStaticSlot is set to STATIC-SLOT-DURATION.</p> <p>FrIfGdSymbolWindow is set to SYMBOL-WINDOW.</p> <p>FrIfGdTSSTransmitter is set to TRANSMISSION-START-SEQUENCE-DURATION.</p> <p>FrIfGListenNoise is set to LISTEN-NOISE.</p> <p>FrIfGMacroPerCycle is set to MACRO-PER-CYCLE.</p> <p>FrIfGdMacrotick is set to MACROTICK-DURATION.</p> <p>FrIfGMaxWithoutClockCorrectPassive is set to MAX-WITHOUT-CLOCK-CORRECTION-PASSIVE.</p> <p>FrIfGMaxWithoutClockCorrectFatal is set to MAX-WITHOUT-CLOCK-CORRECTION-FATAL.</p> <p>FrIfGNumberOfMinislots is set to NUMBER-OF-MINISLOTS.</p>

Configuration parameters	Mapping description
	<p><b>FrIfGChannels</b> is set depending on the FLEXRAY-CLUSTER's PHYSICAL-CHANNEL elements to which the imported ECU-INSTANCE is connected:</p> <ul style="list-style-type: none"> <li>▶ FR_CHANNEL_A if connected to CHANNEL-A.</li> <li>▶ FR_CHANNEL_B if connected to CHANNEL-B.</li> <li>▶ FR_CHANNEL_AB if connected to both CHANNEL-A and CHANNEL-B.</li> </ul> <p><b>FrIfGdSampleClockPeriod</b> is set depending on the value of SAMPLE-CLOCK-PERIOD[s]:</p> <ul style="list-style-type: none"> <li>▶ T50NS for values <math>\geq 0.000000050</math>.</li> <li>▶ T25S for values <math>\geq 0.000000025</math> and <math>&lt; 0.000000050</math>.</li> <li>▶ T12_5NS for values <math>\geq 0.0000000125</math> and <math>&lt; 0.000000025</math>.</li> <li>▶ For values <math>&lt; 0.0000000125</math> <b>FrIfGdSampleClockPeriod</b> is set to T12_5NS and a warning is reported.</li> </ul> <p><b>FrIfGdBit</b> is set depending on the value of BIT [s], BAUDRATE [bit/s] or SPEED [kbit/s] according to the following formulas:</p> <ul style="list-style-type: none"> <li>▶ <math>BIT * 10^9</math>.</li> <li>▶ <math>10^9 / \text{BAUDRATE}[\text{bit/s}]</math> if BIT is not available.</li> <li>▶ <math>10^6 / \text{SPEED}[\text{kbit/s}]</math> if neither BIT nor BAUDRATE are available.</li> </ul> <p><b>FrIfGdBit</b> is then configured depending on the calculation result:</p> <ul style="list-style-type: none"> <li>▶ T100NS for 100.</li> <li>▶ T200NS for 200.</li> <li>▶ T400NS for 400.</li> <li>▶ For all other values <b>FrIfGdBit</b> is set to T100NS and a warning is reported.</li> </ul> <p><b>FrIfDetectNITError</b> is set to DETECT-NIT-ERROR.</p> <p><b>FrIfGCycleCountMax</b> is set to CYCLE-COUNT-MAX.</p> <p><b>FrIfGdIgnoreAfterTx</b> is set to IGNORE-AFTER-TX.</p> <p><b>FrIfGdSymbolWindowActionPointOffset</b> is set to SYMBOL-WINDOW-ACTION-POINT-OFFSET.</p> <p><b>FrIfGdWakeupTxActive</b> is set to WAKEUP-TX-ACTIVE.</p>

Configuration parameters	Mapping description
	<p><code>FrIfGSyncFrameIDCountMax</code> is set to <code>SYNC-FRAME-ID-COUNT-MAX</code>.</p> <p><code>FrIfSafetyMargin</code> is set to <code>SAFETY-MARGIN</code>.</p>
<code>FrIfConfig/FrIf-Cluster/FrIfController</code>	<p>For every <code>FLEXRAY-COMMUNICATION-CONTROLLER</code> connected to the <code>FLEXRAY-CLUSTER</code> and the imported <code>ECU-INSTANCE</code> a <code>FrIfController</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>FLEXRAY-COMMUNICATION-CONTROLLER</code>.</p> <p><code>FrIfFrCtrlRef</code> references the corresponding container in the <code>Fr</code> module configuration.</p>
<code>FrIfConfig/FrIf-Cluster/FrIf-Controller/FrIf-FrameTriggering</code>	<p>For every <code>FLEXRAY-FRAME-TRIGGERING</code> which contains via the referenced <code>FLEXRAY-FRAME</code> and <code>I-PDU-TO-FRAME-MAPPING</code> elements PDUs sent or received by the imported <code>ECU-INSTANCE</code> a <code>FrIfFrameTriggering</code> container is created. The container name is <code>t&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>FLEXRAY-FRAME</code>.</p> <p><code>FrIfChannel</code> is set depending on which <code>PHYSICAL-CHANNEL</code> the <code>FLEXRAY-FRAME-TRIGGERING</code> is sent/received:</p> <ul style="list-style-type: none"> <li>▶ <code>FRIF_CHANNEL_A</code> if sent/received on <code>CHANNEL-A</code>.</li> <li>▶ <code>FRIF_CHANNEL_B</code> if sent/received on <code>CHANNEL-B</code>.</li> <li>▶ <code>FRIF_CHANNEL_AB</code> if two <code>FLEXRAY-FRAME-TRIGGERING</code> elements exist, one on <code>CHANNEL-A</code>, the other on <code>CHANNEL-B</code>, which reference the same <code>FLEXRAY-FRAME</code>. In this case the following conditions must hold as well: <ul style="list-style-type: none"> <li>▶ <code>BASE-CYCLE</code>, <code>CYCLE-REPETITION</code>, and <code>SLOT-ID</code> must be identical.</li> <li>▶ Both <code>FLEXRAY-FRAME-TRIGGERING</code> must have the same transmission direction (sent/received).</li> <li>▶ The <code>SLOT-ID</code> must be <code>&lt;= NUMBER-OF-STATIC-SLOTS</code> as defined in the <code>FLEXRAY-CLUSTER</code>.</li> <li>▶ <code>ALLOW-DYNAMIC-L-SDU-LENGTH</code>, <code>PAYLOAD-PREAMBLE-INDICATOR</code>, and <code>MESSAGE-ID</code> must be identical or undefined for both <code>FLEXRAY-FRAME-TRIGGERING</code> elements.</li> </ul> </li> </ul> <p>If the conditions hold, only a single <code>FrIfFrameTriggering</code> container is created.</p> <p><code>FrIfFrameStructureRef</code> references the container created for the <code>FLEXRAY-FRAME</code> referenced by the <code>FLEXRAY-FRAME-TRIGGERING</code> (see <code>FrIfConfig/FrIfFrameStructure</code>).</p>

Configuration parameters	Mapping description
	<p>FrIfBaseCyle, FrIfCycleRepetition are set to BASE-CYCLE and CYCLE-REPETITION of the FLEXRAY-FRAME-TRIGGERING elements FLEXRAY-ABSOLUTELY-SCHEDULED-TIMING respectively, where values of multiple FLEXRAY-ABSOLUTELY-SCHEDULED-TIMING elements are normalized (i.e. BASE-CYCLE-0/CYCLE-REPETITION-2 and BASE-CYCLE-1/CYCLE-REPETITION-2 are normalized to BASE-CYCLE-0/CYCLE-REPETITION-1).</p> <p>FrIfLSduLength is set to FRAME-LENGTH of the FLEXRAY-FRAME</p> <p>FrIfSlotId is set to SLOT-ID.</p> <p>FrIfPayloadPreamble is set to PAYLOAD-PREAMBLE-INDICATOR.</p> <p>FrIfAllowDynamicLSduLength is set to ALLOW-DYNAMIC-L-SDU-LENGTH.</p> <p>FrIfMessageId is set to MESSAGE-ID.</p>
FrIfConfig/FrIf-FrameStructure	<p>For every FRAME instance sent/received by the imported ECU-INSTANCE a FrIfFrameStructure container is created. The container name is f&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the FLEXRAY-FRAME.</p>
FrIfConfig/FrIf-FrameStructure/FrIfPduIn-Frame	<p>For every PDU sent/received by the imported ECU-INSTANCE and contained by the FLEXRAY-FRAME for which the FrIfFrameStructure container has been created, a FrIfPduInFrame container is created. The container name is p&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>FrIfPduOffset is set to START-POSITION of the PDU-TO-FRAME-MAPPING, divided by 8.</p> <p>FrIfPduUpdateBitOffset is set to UPDATE-INDICATION-BIT-POSITION of the PDU-TO-FRAME-MAPPING, converted to monotone representation.</p> <p>FrIfPduRef references the container created for this PDU under FrIfConfig/FrIfPdu.</p>
FrIfConfig/FrIfPdu	<p>For every PDU instance (see <a href="#">Section 3.3.1.3, "Instance handling"</a>) sent, received or routed (see <a href="#">Section 3.3.6, "PDU routing"</a>) by the imported ECU-INSTANCE, a FrIfPdu container is created. The container name is p&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU belonging to the PDU instance.</p>



Configuration parameters	Mapping description
	<p><b>FrIfPduDirection</b> is set to <b>FrIfRxPdu</b> if received by the imported ECU-INSTANCE, or to <b>FrIfTxPdu</b> if sent by the imported ECU-INSTANCE. The name of <b>FrIfRxPdu</b>/<b>FrIfTxPdu</b> is set to <b>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</b>, where <b>&lt;name&gt;</b> is the <b>SHORT-NAME</b> of the PDU belonging to the PDU instance.</p> <p><b>FrIfRxPduRef</b>/<b>FrIfTxPduRef</b> references the corresponding container in the <b>EcuC</b> module configuration.</p> <p><b>FrIfUserTxUL</b>/<b>FrIfUserRxIndicationUL</b> is set depending on the type of PDU:</p> <ul style="list-style-type: none"> <li>▶ <b>FR_NM</b> for NM-PDU elements.</li> <li>▶ <b>FR_TP</b> for N-PDU elements that are not routed via gateway.</li> <li>▶ <b>PDUR</b> for I-SIGNAL-PDU elements, MULTIPLEXED-I-PDU elements, CONTAINER-I-PDU elements, GENERAL-PURPOSE-I-PDU elements, DCM-I-PDU elements, for USER-DEFINED-I-PDU elements, which either have a <b>CATEGORY</b> other than <b>XCP</b> or are routed, and for N-PDU elements that are routed (see <a href="#">Section 3.3.6, “PDU routing”</a>).</li> <li>▶ <b>XCP</b> for XCP-PDU elements.</li> <li>▶ <b>CDD</b> for USER-DEFINED-I-PDU elements that have their <b>CATEGORY</b> set to <b>XCP</b> and for all other PDU types.</li> </ul>

### 3.4.25. FrNm

Configuration parameters	Mapping description
<b>FrNmGlobalConfig</b> / <b>FrNmGlobalProperties</b>	<p>The following parameters are set using the first <b>FLEXRAY-NM-ECU</b> of all <b>FLEXRAY-NM-CLUSTER</b> elements belonging to the imported ECU-INSTANCE:</p> <p><b>FrNmMainAcrossFrCycle</b> is set to <b>NM-MAIN-FUNCTION-ACROSS-FR-CYCLE</b>.</p>
<b>FrNmGlobalConfig</b> / <b>FrNmGlobalFeatures</b>	<p><b>FrNmComUserDataSupport</b> is set to <b>true</b> if <b>FrNmRxUserDataPduRef</b> or <b>FrNmTxUserDataPduRef</b> is set for any <b>FrNmChannelConfig</b>/<b>FrNmChannelIdentifiers</b>, or if any <b>FLEXRAY-NM-CLUSTER</b> linked to the imported ECU-INSTANCE has its <b>NM-PNC-PARTICIPATION</b> either not defined or set to <b>true</b>. Otherwise <b>FrNmComUserDataSupport</b> is set to <b>false</b>.</p>

Configuration parameters	Mapping description
	<p><code>FrNmPnResetTime</code> is set to <code>PN-RESET-TIME</code> of the configured <code>ECU-INSTANCE</code>.</p> <p>The following parameters are set using the first <code>NM-ECU</code> of all <code>FLEXRAY-NM-CLUSTER</code> elements belonging to the imported <code>ECU-INSTANCE</code>:</p> <p><code>FrNmBusSynchronizationEnabled</code> is set to <code>NM-BUS-SYNCHRONIZATION-ENABLED</code>.</p> <p><code>FrNmDualChannelPduEnable</code> is set to <code>NM-MULTIPLE-CHANNELS-ENABLED</code>.</p> <p><code>FrNmPassiveModeEnabled</code>: see <a href="#">Section 3.4.4, “CanNm”</a>, <code>CanNmPassiveModeEnabled</code>.</p> <p><code>FrNmPduRxIndicationEnabled</code> is set to <code>NM-PDU-RX-INDICATION-ENABLED</code>.</p> <p><code>FrNmRemoteSleepIndicationEnabled</code> is set to <code>NM-REMOTE-SLEEP-ENABLED</code>.</p> <p><code>FrNmStateChangeIndicationEnabled</code> is set to <code>NM-STATE-CHANGE-ENABLED</code>.</p> <p><code>FrNmUserDataEnabled</code> is set to <code>NM-USER-DATA-ENABLED</code>.</p> <p><code>FrNmRepeatMessageBitEnabled</code> is set to <code>NM-REPEAT-MESSAGE-BIT-ENABLE</code>.</p> <p>If inconsistencies are detected among parameters of multiple <code>NM-ECU</code> elements, a warning is reported.</p> <p><code>FrNmHwVoteEnable</code> is set to <code>NM-HW-VOTE-ENABLED</code> of the <code>FLEXRAY-NM-ECU</code> belonging to the <code>NM-ECU</code>.</p> <p><code>FrNmControlBitVectorEnabled</code> is set to <code>NM-CONTROL-BIT-VECTOR-ENABLED</code> of the first <code>FLEXRAY-NM-CLUSTER-COUPPING</code> belonging to the imported <code>ECU-INSTANCE</code>.</p>
<code>FrNmChannelConfig/FrNmChannel</code>	<p>For every <code>FLEXRAY-NM-CLUSTER</code> which belongs to the imported <code>ECU-INSTANCE</code>, a <code>FrNmChannel</code> container is created.</p> <p>The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>FLEXRAY-NM-CLUSTER</code>.</p>

Configuration parameters	Mapping description
	<p>A FLEXRAY-NM-CLUSTER belongs to the imported ECU-INSTANCE if at least one of the FLEXRAY-NM-CLUSTER's FLEXRAY-NM-NODE elements references a FLEXRAY-COMMUNICATION-CONTROLLER of this ECU-INSTANCE.</p> <p>The NM-ECU used for configuring some of the FrNmChannel parameters is the NM-ECU which the first FLEXRAY-NM-NODE references via NM-IF-ECU-REF.</p>
FrNmChannelConfig/FrNmChannel/FrNmChannelIdentifiers	<p>FrNmNodeDetectionEnabled is set to NM-CLUSTER/NM-NODE-DETECTION-ENABLED, or to NM-ECU/NM-NODE-DETECTION-ENABLED if NM-CLUSTER/NM-NODE-DETECTION-ENABLED is not available.</p> <p>FrNmSourceNodeIdIdentifierEnabled is set to NM-CLUSTER/NM-NODE-ID-ENABLED, or to NM-ECU/NM-NODE-ID-ENABLED if NM-CLUSTER/NM-NODE-ID-ENABLED is not available.</p> <p>FrNmControlBitVectorActive is set to NM-CONTROL-BIT-VECTOR-ACTIVE.</p> <p>FrNmRepeatMessageBitActive is set to NM-REPEAT-MESSAGE-BIT-ACTIVE.</p> <p>FrNmSynchronizationPointEnabled is set to NM-SYNCHRONIZING-NETWORK.</p> <p>FrNmChannelHandle references the FrIfCluster container in the FrIf module configuration, which has been created for the FLEXRAY-CLUSTER referenced via COMMUNICATION-CLUSTER-REF.</p> <p>FrNmNodeId is set to NM-NODE-ID of the first FLEXRAY-NM-NODE which belongs to the FLEXRAY-NM-CLUSTER and at the same time belongs to the ECU-INSTANCE. A FLEXRAY-NM-NODE belongs to an ECU-INSTANCE if it references a FLEXRAY-COMMUNICATION-CONTROLLER of this ECU-INSTANCE.</p> <p>FrNmComMNetworkHandleRef references the ComMChannel container that is created for the COMMUNICATION-CLUSTER referenced in COMMUNICATION-CLUSTER-REF.</p> <p>FrNmPduScheduleVariant is set depending on the value of NM-SCHEDULE-VARIANT of the first FLEXRAY-NM-CLUSTER-COUPLING belonging to the FLEXRAY-NM-CLUSTER:</p> <p>► FRNM_PDU_SCHEDULE_VARIANT_&lt;x&gt; is set to SCHEDULE-VARIANT-&lt;x&gt; where x lies within [1..7].</p>

Configuration parameters	Mapping description
FrNmChannelConfig/FrNmChannelTiming	<p>FrNmDataCycle is set to FRNM_CYCLE_VALUE_&lt;value of NM-DATA-CYCLE&gt;. If NM-DATA-CYCLE is not one of [1,2,4,8,18,32,64], a warning is reported and FrNmDataCycle is not configured.</p> <p>FrNmRepetitionCycle is set depending on the value of NM-REPETITION-CYCLE, see FrNmDataCycle for how values are mapped.</p> <p>FrNmVotingCycle is set depending on the value of NM-VOTING-CYCLE, see FrNmDataCycle for how values are mapped.</p> <p>FrNmMainFunctionPeriod is set to NM-MAIN-FUNCTION-PERIOD.</p> <p>FrNmMsgTimeoutTime is set to NM-MESSAGE-TIMEOUT-TIME.</p> <p>FrNmNodeDetectionLock is set to NM-DETECTION-LOCK.</p> <p>FrNmReadySleepCnt is calculated by the formula <math>(\text{NM-READY-SLEEP-TIME} / \text{CYCLE}) / \text{NM-REPETITION-CYCLE} - 1</math>. NM-READY-SLEEP-TIME is taken from the first FLEXRAY-COMMUNICATION-CONNECTOR connecting the COMMUNICATION-CONTROLLER of the first FLEXRAY-NM-NODE to a PHYSICAL-CHANNEL of the FLEXRAY-NM-CLUSTER's FLEXRAY-COMMUNICATION-CLUSTER. A warning is reported if multiple different NM-READY-SLEEP-TIME values are found. If the calculation yields a non-integral value, a warning is reported and FrNmReadySleepCnt is not set.</p> <p>If NM-READY-SLEEP-TIME is not available, FrNmReadySleepCnt is set to NM-READY-SLEEP-COUNT.</p> <p>FrNmRemoteSleepIndTime is set to NM-REMOTE-SLEEP-INDICATION-TIME.</p> <p>FrNmRepeatMessageTime is set to NM-REPEAT-MESSAGE-TIME.</p>
FrNmChannelConfig/FrNmChannelIdentifiers/FrNmRxPdu	<p>For every NM-PDU referenced via RX-NM-PDU-REF by the first FLEXRAY-NM-NODE belonging to the FLEXRAY-NM-CLUSTER and the imported ECU-INSTANCE, a FrNmRxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the NM-PDU.</p> <p>If either none or more than two NM-PDU elements are referenced via RX-NM-PDU-REF a warning is reported.</p> <p>FrNmRxPduContainsData is set to true if the NM-PDU contains I-SIGNAL-TO-I-PDU-MAPPING elements referencing an I-SIGNAL or an I-SIG-</p>

Configuration parameters	Mapping description
	<p>NAL-GROUP, or if NM-DATA-INFORMATION is set to true. In all other cases FrNmRxPduContainsData is set to false.</p> <p>FrNmRxPduPduContainsVote is set to NM-VOTE-INFORMATION.</p> <p>FrNmRxPduRef references the corresponding container in the EcuC module configuration.</p>
FrNmChannelConfig/FrNmChannel/FrNmChannelIdentifiers/FrNmUserDataRxPdu	<p>The list of all NM-PDU elements that are referenced via RX-NM-PDU-REF by the first FLEXRAY-NM-NODE that belongs to the FLEXRAY-NM-CLUSTER and the imported ECU-INSTANCE constitutes the received NM-PDU elements of the FLEXRAY-NM-CLUSTER.</p> <p>The FrNmUserDataRxPdu container is created if an NmUserDataPdu container has been created in the EcuC module configuration for at least one of these NM-PDU elements, see <a href="#">Section 3.4.14, "EcuC"</a>.</p> <p>The name of FrNmUserDataRxPdu is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;_NmComUserData, where &lt;name&gt; is the SHORT-NAME of the NM-PDU.</p> <p>FrNmRxUserDataPduRef references the NmUserDataPdu container in the EcuC module which has been created for the NM-PDU. If more than one NM-PDU contains I-SIGNAL-TO-I-PDU-MAPPING elements referencing an I-SIGNAL or an I-SIGNAL-GROUP, the first NM-PDU is used and a warning is reported.</p>
FrNmChannelConfig/FrNmChannel/FrNmChannelIdentifiers/FrNmTxPdu	<p>For every NM-PDU referenced via TX-NM-PDU-REF by the first FLEXRAY-NM-NODE belonging to the FLEXRAY-NM-CLUSTER and the imported ECU-INSTANCE, a FrNmTxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the NM-PDU.</p> <p>If more than two NM-PDU elements are referenced via TX-NM-PDU-REF a warning is reported.</p> <p>FrNmTxPduContainsData is set to true if the NM-PDU contains an I-SIGNAL-TO-I-PDU-MAPPING referencing an I-SIGNAL or an I-SIGNAL-GROUP, or if NM-DATA-INFORMATION is set to true. In all other cases FrNmTxPduContainsData is set to false.</p> <p>FrNmTxPduPduContainsVote is set to NM-VOTE-INFORMATION.</p> <p>FrNmTxPduRef references the corresponding container in the EcuC module configuration.</p>

Configuration parameters	Mapping description
FrNmChannelConfig/FrNmChannel/FrNmChannelIdentifiers/FrNmUserDataTxPdu	See FrNmUserDataRxPdu (NM-PDU elements referenced via TX-NM-PDU-REF are used for FrNmUserDataTxPdu).

For the configuration of PNC-related parameters see [Section 3.4.4, “CanNm”](#). The parameters and configuration containers obtain the prefix `FrNmPn`. The `PNC-FILTER-DATA-MASK` values of all `FLEXRAY-COMMUNICATION-CONNECTOR` elements that belong to the imported `ECU-INSTANCE` are taken as input for the calculation of `FrNmPnFilterMaskByte`.

### 3.4.26. FrSM

Configuration parameters	Mapping description
FrSMConfig/FrSMCluster	<p>For every <code>FLEXRAY-CLUSTER</code> which belongs to the imported <code>ECU-INSTANCE</code>, a <code>FrSMCluster</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>FLEXRAY-CLUSTER</code>.</p> <p><code>FrSMComMNetworkHandleRef</code> references the <code>ComMChannel</code> container in the <code>ComM</code> module configuration, which has been created for the <code>FLEXRAY-CLUSTER</code>.</p> <p><code>FrSMFrIfClusterRef</code> references the <code>FrIfCluster</code> container in the <code>FrIf</code> module configuration, which has been created for the <code>FLEXRAY-CLUSTER</code>.</p>

### 3.4.27. FrTp

Configuration parameters	Mapping description
FrTpGeneral	<p>The following parameters are set using the first <code>FLEXRAY-TP-ECU</code> belonging to the imported <code>ECU-INSTANCE</code>:</p> <p><code>FrTpMainFuncCycle</code> is set to <code>CYCLE-TIME-MAIN-FUNCTION</code>.</p> <p><code>FrTpFullDuplexEnable</code> is set to <code>FULL-DUPLEX-ENABLED</code>.</p> <p><code>FrTpTransmitCancellation</code> is set to <code>CANCELLATION</code>.</p>

Configuration parameters	Mapping description
	If inconsistencies are detected among parameters of multiple FLEXRAY-TP-ECU elements of the imported ECU-INSTANCE, a warning is reported.
FrTpMultipleConfig/FrTpConnection	<p>For each FLEXRAY-TP-CONNECTION which belongs to the imported ECU-INSTANCE, one FrTpConnection container is created, named FrTpConnection_&lt;suffix&gt;, where &lt;suffix&gt; is a zero-based index. A FLEXRAY-TP-CONNECTION belongs to the imported ECU-INSTANCE if at least one transmitting or receiving FLEXRAY-TP-NODE of this FLEXRAY-TP-CONNECTION references a COMMUNICATION-CONNECTOR which is also referenced by the imported ECU-INSTANCE.</p> <p>Two FLEXRAY-TP-CONNECTION elements &lt;FrTp1&gt; and &lt;FrTp2&gt; are configured as one single FrTpConnection container if they both belong to the imported ECU-INSTANCE and the following conditions hold:</p> <ul style="list-style-type: none"> <li>▶ &lt;FrTp1&gt; and &lt;FrTp2&gt; both refer to exactly one FLEXRAY-TP-NODE via RECEIVER-REFS</li> <li>▶ The single FLEXRAY-TP-NODE in RECEIVER-REFS of &lt;FrTp1&gt; is the same as the one referenced in TRANSMITTER-REF of &lt;FrTp2&gt; and vice versa.</li> <li>▶ Both &lt;FrTp1&gt; and &lt;FrTp2&gt; refer to the exactly one PDU, either via DIRECT-TP-SDU-REF, or via REVERSED-TP-SDU-REF</li> <li>▶ Neither &lt;FrTp1&gt; nor &lt;FrTp2&gt; refers to a TP-ADDRESS via MULTI-CAST-REF</li> <li>▶ Both &lt;FrTp1&gt; and &lt;FrTp2&gt; refer to the same FLEXRAY-TP-CONNECTION-CONTROL via TP-CONNECTION-CONTROL-REF</li> <li>▶ Both &lt;FrTp1&gt; and &lt;FrTp2&gt; either have the same BANDWIDTH-LIMITATION value configured, or none of &lt;FrTp1&gt; and &lt;FrTp2&gt; has that value configured</li> </ul> <p>In that case, the configuration values of &lt;FrTp1&gt; are used to configure the FrTpConnection container. Additionally, the PDU of &lt;FrTp2&gt; is configured in the container as well, either in FrTpRxSdu or in FrTpTxSdu.</p> <p>FrTpConCtrlRef references the container created for the FLEXRAY-TP-CONNECTION-CONTROL referenced (see FrTpMultipleConfig/FrTpConnectionControl).</p> <p>FrTpRxPduPoolRef references the container created for the Rx N-PDU elements referenced via FLEXRAY-TP-PDU-POOL elements (see FrTpMultipleConfig/FrTpRxPduPool).</p>

Configuration parameters	Mapping description
	<p><b>FrTpTxPduPoolRef</b> references the container created for the Tx N-PDU elements referenced via FLEXRAY-TP-PDU-POOL elements (see FrTpMultipleConfig/FrTpTxPduPool).</p> <p><b>FrTpRxSdu</b>: this container is created if the PDU referenced by the FLEXRAY-TP-CONNECTION via DIRECT-TP-SDU-REF is received or routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) by the imported ECU-INSTANCE, or if this PDU is referenced via REVERSED-TP-SDU-REF and is sent or routed by the imported ECU-INSTANCE. The name of FrTpRxSdu is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p><b>FrTpRxSdu/FrTpRxSduRef</b> references the corresponding container in the EcuC module configuration.</p> <p><b>FrTpTxSdu</b>: this container is created if the PDU referenced by the FLEXRAY-TP-CONNECTION via DIRECT-TP-SDU-REF is sent or routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) by the imported ECU-INSTANCE, or if this PDU is referenced via REVERSED-TP-SDU-REF and is received or routed by the imported ECU-INSTANCE. The name of FrTpTxSdu is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p><b>FrTpTxSdu/FrTpTxSduRef</b> references the corresponding container in the EcuC module configuration.</p> <p><b>FrTpBandwidthLimitation</b> is set to BANDWIDTH-LIMITATION.</p> <p><b>FrTpMultipleReceiverCon</b> is set to true if a valid TP-ADDRESS is referenced by the FLEXRAY-TP-CONNECTION via MULTICAST-REF. Otherwise FrTpMultipleReceiverCon is set to false.</p> <p><b>FrTpLa</b>: if the FLEXRAY-TP-NODE referenced via TRANSMITTER-REF belongs to the imported ECU-INSTANCE, FrTpLa is set to the TP-ADDRESS referenced by this FLEXRAY-TP-NODE. Otherwise FrTpLa is set to the TP-ADDRESS referenced by the FLEXRAY-TP-CONNECTION via MULTICAST-REF, or, if this is not available, to the TP-ADDRESS of the first FLEXRAY-TP-NODE referenced via RECEIVER-REF.</p> <p><b>FrTpRa</b>: if the FLEXRAY-TP-NODE referenced via TRANSMITTER-REF does not belong to the imported ECU-INSTANCE, FrTpRa is set to the TP-ADDRESS referenced by this FLEXRAY-TP-NODE. Otherwise FrTpRa is set to the TP-ADDRESS referenced by the FLEXRAY-TP-CONNECTION via MULTICAST-REF, or,</p>



Configuration parameters	Mapping description
	if this is not available, to the TP-ADDRESS of the first FLEXRAY-TP-NODE referenced via RECEIVER-REF.
FrTpMultipleConfig/FrTpConnectionControl	<p>For each FLEXRAY-TP-CONNECTION-CONTROL referenced by an imported FLEXRAY-TP-CONNECTION a FrTpConnectionControl container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the FLEXRAY-TP-CONNECTION-CONTROL.</p> <p>FrTpAckType is set depending on the value of ACK-TYPE:</p> <ul style="list-style-type: none"> <li>▶ FRTP_ACK_WITH_RT for ACK-WITH-RT.</li> <li>▶ FRTP_NO for NO-ACK.</li> </ul> <p>FrTpMaxAr is set to MAX-AR.</p> <p>FrTpMaxAs is set to MAX-AS.</p> <p>FrTpMaxBufferSize is set to MAX-BUFFER-SIZE.</p> <p>FrTpMaxFCWait is set to MAX-FC-WAIT.</p> <p>FrTpMaxFrIf is set to MAX-FR-IF.</p> <p>FrTpMaxRn is set to MAX-RETRIES.</p> <p>FrTpTimeBr is set to TIME-BR.</p> <p>FrTpTimeBuffer is set to TIME-BUFFER.</p> <p>FrTpTimeFrIf is set to TIME-FR-IF.</p> <p>FrTpTimeoutAr is set to TIMEOUT-AR.</p> <p>FrTpTimeoutAs is set to TIMEOUT-AS.</p> <p>FrTpTimeoutBs is set to TIMEOUT-BS.</p> <p>FrTpTimeoutCr is set to TIMEOUT-CR.</p> <p>FrTpSCexp is set to SEPARATION-CYCLE-EXPONENT.</p> <p>FrTpMaxNbrOfNPduPerCycle is set to MAX-NUMBER-OF-NPDU-PER-CYCLE.</p>
FrTpMultipleConfig/FrTpRxPduPool	For each FLEXRAY-TP-PDU-POOL that is referenced by a sent FLEXRAY-TP-CONNECTION via RX-PDU-POOL-REF or that is referenced by a received FLEXRAY-TP-CONNECTION via TX-PDU-POOL-REF, and which contains only references to N-PDU elements received by the imported ECU-INSTANCE, an

Configuration parameters	Mapping description
	<p><b>FrTpRxPduPool</b> container is created. A <b>FLEXRAY-TP-CONNECTION</b> is considered sent/received if it belongs to the imported <b>ECU-INSTANCE</b> and if it references a PDU via <b>DIRECT-TP-SDU-REF</b> which is sent/received by this <b>ECU-INSTANCE</b>. The container name is <b>&lt;name&gt;_Rx</b>, where <b>&lt;name&gt;</b> is the <b>SHORT-NAME</b> of the <b>FLEXRAY-TP-PDU-POOL</b>.</p> <p>If a <b>FLEXRAY-TP-PDU-POOL</b> that is referenced by a sent <b>FLEXRAY-TP-CONNECTION</b> via <b>RX-PDU-POOL-REF</b> or that is referenced by a received <b>FLEXRAY-TP-CONNECTION</b> via <b>TX-PDU-POOL-REF</b>, contains any references to PDUs sent by the imported <b>ECU-INSTANCE</b>, the following algorithm is used to create <b>FrTpRxPduPool</b> containers:</p> <p>For each unique set of <b>N-PDU</b> elements received by the imported <b>ECU-INSTANCE</b> and referenced by <b>FLEXRAY-TP-PDU-POOL</b> elements of a <b>FLEXRAY-TP-CONNECTION</b> (referenced via both <b>RX-PDU-POOL-REF</b> and <b>TX-PDU-POOL-REF</b>), a <b>FrTpRxPduPool</b> container named <b>PduPool_&lt;suffix&gt;</b> is created, where <b>&lt;suffix&gt;</b> is a zero-based index.</p> <p>For all referenced <b>N-PDU</b> elements the parameter <b>FrTpRxPduRef</b> is set (using a reference to corresponding container in the <b>EcuC</b> module).</p>
FrTpMultipleConfig/FrTpTxPduPool	<p>For each <b>FLEXRAY-TP-PDU-POOL</b> that is referenced by a sent <b>FLEXRAY-TP-CONNECTION</b> via <b>TX-PDU-POOL-REF</b> or that is referenced by a received <b>FLEXRAY-TP-CONNECTION</b> via <b>RX-PDU-POOL-REF</b>, and which contains only references to <b>N-PDU</b> elements sent by the imported <b>ECU-INSTANCE</b>, an <b>FrTpTxPduPool</b> container is created. The container name is <b>&lt;name&gt;_Tx</b>, where <b>&lt;name&gt;</b> is the <b>SHORT-NAME</b> of the <b>FLEXRAY-TP-PDU-POOL</b>.</p> <p>If a <b>FLEXRAY-TP-PDU-POOL</b> that is referenced by a sent <b>FLEXRAY-TP-CONNECTION</b> via <b>TX-PDU-POOL-REF</b> or that is referenced by a received <b>FLEXRAY-TP-CONNECTION</b> via <b>RX-PDU-POOL-REF</b>, contains any references to PDUs received by the imported <b>ECU-INSTANCE</b>, the following algorithm is used to create <b>FrTpRxPduPool</b> containers:</p> <p>For each unique set of <b>N-PDU</b> elements sent by the imported <b>ECU-INSTANCE</b> and referenced by <b>FLEXRAY-TP-PDU-POOL</b> elements of a <b>FLEXRAY-TP-CONNECTION</b> (referenced via both <b>RX-PDU-POOL-REF</b> and <b>TX-PDU-POOL-REF</b>), a <b>FrTpTxPduPool</b> container named <b>PduPool_&lt;suffix&gt;</b> is created, where <b>&lt;suffix&gt;</b> is a zero-based index.</p> <p>For all referenced <b>N-PDU</b> elements the parameter <b>FrTpTxPduRef</b> is set (using a reference to corresponding container in the <b>EcuC</b> module).</p>

### 3.4.28. FrTSyn

Configuration parameters	Mapping description
FrTSynGlobalTimeDomain	<p>One FrTSynGlobalTimeDomain container is created for each GLOBAL-TIME-DOMAIN that contains a GLOBAL-TIME-FR-MASTER or a GLOBAL-TIME-FR-SLAVE element that references a FLEXRAY-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-DOMAIN.</p> <p>FrTSynGlobalTimeDomainId is set to DOMAIN-ID.</p> <p>FrTSynSynchronizedTimeBaseRef references the corresponding StbMSynchronizedTimeBase container in the StbM module. If the GLOBAL-TIME-DOMAIN is a subdomain of a parent GLOBAL-TIME-DOMAIN, FrTSynSynchronizedTimeBaseRef references the StbMSynchronizedTimeBase container that has been created for the parent GLOBAL-TIME-DOMAIN. For further information about the configuration of the StbM module, see <a href="#">Section 3.4.45, “StbM”</a>.</p>
FrTSynGlobalTimeDomain/FrTSynGlobalTimeOfsDataIDList, FrTSynGlobalTimeDomain/FrTSynGlobalTimeSyncDataIDList	<p>The FR-GLOBAL-TIME-DOMAIN-PROPS element of the FlexRay GLOBAL-TIME-DOMAIN entity is used to retrieve the following subelements to configure container lists:</p> <ul style="list-style-type: none"> <li>▶ OFS-DATA-ID-LIST</li> <li>▶ SYNC-DATA-ID-LIST</li> </ul> <p>The configured container lists are the following:</p> <ul style="list-style-type: none"> <li>▶ FrTSynGlobalTimeDomain/FrTSynGlobalTimeOfsDataIDList/FrTSynGlobalTimeOfsDataIDListElement</li> <li>▶ FrTSynGlobalTimeDomain/FrTSynGlobalTimeSyncDataIDList/FrTSynGlobalTimeSyncDataIDListElement</li> </ul> <p>In each container list, one subcontainer is created per element named Element_&lt;idx&gt;, where &lt;idx&gt; is the zero-based index of the element within the list. In each of these lists, one index parameter and one value parameter are configured. The index parameter represents the zero-based index of the element in the list. The value parameter represents the actual value.</p>

Configuration parameters	Mapping description		
	ID list	Index parameter	Value parameter
	FrTSynGlobalTimeOfsDataIDList	FrTSynGlobalTimeOfsDataIDListIndex	FrTSynGlobalTimeOfsDataIDListValue
	FrTSynGlobalTimeSyncDataIDList	FrTSynGlobalTimeSyncDataIDListIndex	FrTSynGlobalTimeSyncDataIDListValue
FrTSynGlobalTimeDomain/FrTSynGlobalTimeMaster	<p><b>If the GLOBAL-TIME-FR-MASTER of the GLOBAL-TIME-DOMAIN references a FLEXRAY-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE, an FrTSynGlobalTimeMaster container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-FR-MASTER.</b></p> <p>FrTSynGlobalTimeTxCrcSecured is set depending on the value of GLOBAL-TIME-FR-MASTER/CRC-SECURED: CRC_SUPPORTED for CRC-SUPPORTED and CRC_NOT_SUPPORTED for CRC-NOT-SUPPORTED.</p> <p>FrTSynGlobalTimeTxPeriod is set to GLOBAL-TIME-FR-MASTER/SYNC-PERIOD.</p> <p>FrTSynCyclicMsgResumeTime is set to GLOBAL-TIME-FR-MASTER/IMMEDIATE-RESUME-TIME.</p> <p>FrTSynGlobalTimeDebounceTime is set to GLOBAL-TIME-DOMAIN/DEBOUNCE-TIME.</p>		
FrTSynGlobalTimeDomain/FrTSynGlobalTimeMaster/FrTSynGlobalTimeMasterPdu	<p><b>A GLOBAL-TIME-DOMAIN is associated with a PDU if that PDU is a GENERAL-PURPOSE-PDU that has its CATEGORY set to GLOBAL_TIME and one of the conditions holds:</b></p> <ul style="list-style-type: none"> <li>▶ <b>The GLOBAL-TIME-DOMAIN references the PDU in GLOBAL-TIME-PDU-REF</b></li> <li>▶ <b>The GLOBAL-TIME-DOMAIN references a PDU-TRIGGERING either via GLOBAL-TIME-PDU-TRIGGERING-REF or via PDU-TRIGGERING-REF, and that PDU-TRIGGERING refers to the PDU</b></li> </ul> <p><b>If the GLOBAL-TIME-DOMAIN is associated with a PDU, and if this is the PDU that the configured ECU-INSTANCE sends on the FLEXRAY-CLUSTER which the GLOBAL-TIME-DOMAIN references via COMMUNICATION-CLUSTER-REF, one FrTSynGlobalTimeMasterPdu container is created. Its name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</b></p>		

Configuration parameters	Mapping description
	FrTSynGlobalTimePduRef references the EcuC container created for the PDU that is associated with the GLOBAL-TIME-DOMAIN.
FrTSynGlobalTimeDomain/FrTSynGlobalTimeSlave	<p>If one of the GLOBAL-TIME-FR-SLAVE elements of the GLOBAL-TIME-DO-MAIN references a FLEXRAY-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE, an FrTSynGlobalTimeSlave container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-FR-SLAVE.</p> <p>FrTSynRxCrcValidated is set depending on the value of GLOBAL-TIME-FR-SLAVE/CRC-VALIDATED: CRC_VALIDATED for CRC-VALIDATED, CRC_NOT_VALIDATED for CRC-NOT-VALIDATED, and CRC_IGNORED for CRC-IGNORED.</p> <p>FrTSynGlobalTimeSequenceCounterJumpWidth is set to GLOBAL-TIME-FR-SLAVE/SEQUENCE-COUNTER-JUMP-WIDTH.</p>
FrTSynGlobalTimeDomain/FrTSynGlobalTimeSlave/FrTSynGlobalTimeSlavePdu	<p>If the GLOBAL-TIME-DOMAIN is associated with a PDU, and if this is the PDU that the configured ECU-INSTANCE receives on the FLEXRAY-CLUSTER which the GLOBAL-TIME-DOMAIN references via COMMUNICATION-CLUSTER-REF, one FrTSynGlobalTimeSlavePdu container is created. Its name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>FrTSynGlobalTimePduRef references the EcuC container created for the PDU that is associated with the GLOBAL-TIME-DOMAIN.</p>
FrTSynGeneral/FrTSynGlobalTimeSyncDataIDList	<p>For all FlexRay GLOBAL-TIME-DOMAIN entities the configured ECU-INSTANCE is connected to, all FR-GLOBAL-TIME-DOMAIN-PROPS are retrieved. From that FR-GLOBAL-TIME-DOMAIN-PROPS list, the first non-empty SYNC-DATA-ID-LIST is retrieved to configure the container list FrTSynGeneral/FrTSynGlobalTimeSyncDataIDList/FrTSynGlobalTimeSyncDataIDListElement.</p> <p>One subcontainer is created per element named Element_&lt;idx&gt;, where &lt;idx&gt; is the zero-based index of the element within the list. FrTSynGlobalTimeSyncDataIDListIndex contains the zero-based index of the element within the list, FrTSynGlobalTimeSyncDataIDListValue is configured to contain the value of the element.</p>

### 3.4.29. IpduM

Configuration parameters	Mapping description
IpduMGeneral/IpduM-HeaderByteOrder	<p>If the imported ECU-INSTANCE sends or receives at least one CONTAINER-I-PDU, IpduMHeaderByteOrder is configured depending on the CONTAINER-I-PDU-HEADER-BYTE-ORDER parameter of the SYSTEM:</p> <ul style="list-style-type: none"> <li>▶ IPDUM_LITTLE_ENDIAN for MOST-SIGNIFICANT-BYTE-LAST</li> <li>▶ IPDUM_BIG_ENDIAN for MOST-SIGNIFICANT-BYTE-FIRST</li> </ul>
IpduMConfig/IpduM-RxPathway	<p>For every MULTIPLEXED-I-PDU which references at least one PDU in its DYNAMIC-PART-ALTERNATIVE elements or in its STATIC-PART, and which is also received by the imported ECU-INSTANCE, an IpduMRxPathway container is created. The container name is RXP_&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the MULTIPLEXED-I-PDU.</p>
IpduMConfig/IpduM-RxPathway/IpduM-RxIndication	<p>The name of IpduMRxIndication is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the MULTIPLEXED-I-PDU.</p> <p>IpduMRxIndicationPduRef references the container created for the MULTIPLEXED-I-PDU in the EcuC module configuration.</p> <p>IpduMByteOrder is set to LITTLE_ENDIAN.</p> <p>If SELECTOR-FIELD-BYTE-ORDER is set to MOST-SIGNIFICANT-BYTE-FIRST, the bit range defined by SELECTOR-FIELD-START-POSITION and SELECTOR-FIELD-LENGTH is converted to MOST-SIGNIFICANT-BYTE-LAST, which may result in multiple bit ranges. Multiple bit ranges in turn result in an error and the termination of the import. In case the bit range crosses byte boundaries, a warning is reported.</p> <p>IpduMSelectorFieldPosition/IpduMSelectorFieldPosition is set to the start position of the bit range.</p> <p>IpduMSelectorFieldPosition/IpduMSelectorFieldLength is set to the length of the bit range.</p>
IpduMConfig/IpduM-RxPathway/IpduM-RxIndication/IpduM-RxDynamicPart	<p>For every DYNAMIC-PART-ALTERNATIVE of the MULTIPLEXED-I-PDU which references an I-SIGNAL-I-PDU that is routed (see <a href="#">Section 3.3.6, "PDU routing"</a>) or contains at least one I-SIGNAL received by the imported ECU-INSTANCE, an IpduMRxDynamicPart container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-I-PDU.</p>

Configuration parameters	Mapping description
	<p>IpduMOutgoingDynamicPduRef references the container created for the I-SIGNAL-I-PDU in the EcuC module configuration.</p> <p>IpduMRxSelectorValue is set to SELECTOR-FIELD-CODE.</p>
IpduMConfig/IpduMRxPathway/IpduMRxIndication/IpduMRxDynamicPart/IpduMSegment	<p>For every bit range defined by SEGMENT-POSITION elements of the DYNAMIC-PART an IpduMSegment container is created. The container name is IpduMSegment_&lt;auto incremented number&gt;.</p> <p>If SEGMENT-BYTE-ORDER is set to MOST-SIGNIFICANT-BYTE-FIRST, the bit range defined by SEGMENT-POSITION and SEGMENT-LENGTH is converted to MOST-SIGNIFICANT-BYTE-LAST, which may result in multiple bit ranges.</p> <p>IpduMSegmentPosition and IpduMDestinationBit are set to the start position of the bit range.</p> <p>IpduMSegmentLength is set to the length of the bit range.</p>
IpduMConfig/IpduMRxPathway/IpduMRxIndication/IpduMRxStaticPart	<p>If a STATIC-PART of the MULTIPLEXED-I-PDU exists, which references an I-SIGNAL-I-PDU that is routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) or contains at least one I-SIGNAL received by the imported ECU-INSTANCE, an IpduMRxStaticPart container is created. The name of IpduMRxStaticPart is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-I-PDU.</p> <p>IpduMOutgoingStaticPduRef references the container created for the I-SIGNAL-I-PDU in the EcuC module configuration.</p> <p>For the configuration of IpduMSegment refer to IpduMRxDynamicPart/IpduMSegment.</p>
IpduMConfig/IpduMTxPathway	<p>For every MULTIPLEXED-I-PDU sent by the imported ECU-INSTANCE, an IpduMTxPathway container is created. The container name is TXP_&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the MULTIPLEXED-I-PDU.</p>
IpduMConfig/IpduMTxPathway/IpduMTxRequest	<p>The name of IpduMTxRequest is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the MULTIPLEXED-I-PDU.</p> <p>IpduMOutgoingPduRef references the container created for the MULTIPLEXED-I-PDU in the EcuC module configuration.</p> <p>IpduMByteOrder is set to LITTLE_ENDIAN.</p> <p>For the configuration of IpduMSelectorFieldPosition see IpduMRxIndication/IpduMSelectorFieldPosition.</p>

Configuration parameters	Mapping description
	<p><code>IpduMIPduUnusedAreasDefault</code> is set to <code>UNUSED-BIT-PATTERN</code>. If <code>UNUSED-BIT-PATTERN</code> is less than zero or greater than 255, a warning is reported and <code>IpduMIPduUnusedAreasDefault</code> remains undefined.</p> <p><code>IpduMTxTriggerMode</code> is set depending on the value of <code>TRIGGER-MODE</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>STATIC_PART_TRIGGER</code> for <code>STATIC-PART-TRIGGER</code>.</li> <li>▶ <code>DYNAMIC_PART_TRIGGER</code> for <code>DYNAMIC-PART-TRIGGER</code>.</li> <li>▶ <code>STATIC_OR_DYNAMIC_PART_TRIGGER</code> for <code>STATIC-OR-DYNAMIC-PART-TRIGGER</code>.</li> <li>▶ <code>NONE</code> for <code>NONE</code>.</li> </ul> <p><code>IpduMInitialDynamicPart</code> references the <code>IpduMTxDynamicPart</code> container which has been created for the first <code>DYNAMIC-PART-ALTERNATIVE</code> which has its <code>INITIAL-DYNAMIC-PART</code> set to <code>true</code>.</p> <p>If multiple <code>DYNAMIC-PART-ALTERNATIVE</code> elements have its <code>INITIAL-DYNAMIC-PART</code> set to <code>true</code>, a warning is issued.</p>
<p><code>IpduMConfig/IpduMTxPathway/IpduMTxRequest/IpduMTxDynamicPart</code></p>	<p>For every <code>DYNAMIC-PART-ALTERNATIVE</code> of the <code>MULTIPLEXED-I-PDU</code> which references an <code>I-SIGNAL-I-PDU</code> that is routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) or contains at least one <code>I-SIGNAL</code> sent by the imported <code>ECU-INSTANCE</code>, an <code>IpduMTxDynamicPart</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>I-SIGNAL-I-PDU</code>.</p> <p><code>IpduMTxDynamicPduRef</code> references the container created for the <code>I-SIGNAL-I-PDU</code> in the <code>EcuC</code> module configuration.</p> <p><code>IpduMTxSelectorValue</code> is set to <code>SELECTOR-FIELD-CODE</code>.</p> <p><code>IpduMSegment</code>: see <code>IpduMRxDynamicPart/IpduMSegment</code>.</p>
<p><code>IpduMConfig/IpduMTxPathway/IpduMTxRequest/IpduMTxStaticPart</code></p>	<p>If a <code>STATIC-PART</code> of the <code>MULTIPLEXED-I-PDU</code> exists, which references an <code>I-SIGNAL-I-PDU</code> that is routed (see <a href="#">Section 3.3.6, “PDU routing”</a>) or contains at least one <code>I-SIGNAL</code> sent by the imported <code>ECU-INSTANCE</code>, an <code>IpduMTxStaticPart</code> container is created. The name of <code>IpduMTxStaticPart</code> is set to <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>I-SIGNAL-I-PDU</code>.</p> <p><code>IpduMTxStaticPduRef</code> references the container created for the <code>I-SIGNAL-I-PDU</code> in the <code>EcuC</code> module configuration.</p>



Configuration parameters	Mapping description
	For the configuration of IpduMSegment see IpduMRxDynamicPart/IpduMSegment.
IpduMConfig/IpduMContainerRxPdu	<p>For every CONTAINER-I-PDU which is received by the imported ECU-INSTANCE, an IpduMContainerRxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the CONTAINER-I-PDU.</p> <p>IpduMContainerHeaderSize is configured depending on HEADER-TYPE:</p> <ul style="list-style-type: none"> <li>▶ IPDUM_HEADERTYPE_SHORT for SHORT-HEADER</li> <li>▶ IPDUM_HEADERTYPE_LONG for LONG-HEADER</li> <li>▶ IPDUM_HEADERTYPE_NONE for NO-HEADER</li> </ul> <p>IpduMContainerRxAcceptContainedPdu is configured depending on RX-ACCEPT-CONTAINED-I-PDU:</p> <ul style="list-style-type: none"> <li>▶ IPDUM_ACCEPT_ALL for ACCEPT-ALL</li> <li>▶ IPDUM_ACCEPT_CONFIGURED for ACCEPT-CONFIGURED</li> </ul> <p>IpduMContainerRxPduRef references the container created for the CONTAINER-I-PDU in the EcuC module configuration.</p> <p>IpduMContainerQueueSize is set to CONTAINER-I-PDU/MINIMUM-RX-CONTAINER-QUEUE-SIZE if it exists and if it is greater than the pre-existing IpduMContainerQueueSize value.</p>
IpduMConfig/IpduMContainerTxPdu	<p>For every CONTAINER-I-PDU which is sent by the imported ECU-INSTANCE, an IpduMContainerTxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the CONTAINER-I-PDU.</p> <p>IpduMContainerHeaderSize is configured in the same way as in IpduMConfig/IpduMContainerRxPdu.</p> <p>IpduMContainerTxFirstContainedPduTrigger is configured depending on CONTAINER-TRIGGER:</p> <ul style="list-style-type: none"> <li>▶ true for FIRST-CONTAINED-TRIGGER</li> <li>▶ false for DEFAULT-TRIGGER</li> </ul> <p>IpduMContainerTxSendTimeout is set to CONTAINER-TIMEOUT.</p> <p>IpduMContainerTxSizeThreshold is set to THRESHOLD-SIZE.</p>

Configuration parameters	Mapping description
	<p><code>IpduMUnusedAreasDefault</code> is set to <code>UNUSED-BIT-PATTERN</code>.</p> <p><code>IpduMContainerTxPduRef</code> references the container created for the <code>CONTAINER-I-PDU</code> in the <code>EcuC</code> module configuration.</p> <p><code>IpduMContainerTxTriggerMode</code> is set to <code>IPDUM_DIRECT</code> if the <code>CONTAINER-I-PDU</code> is contained in a <code>SECURED-I-PDU</code>. Otherwise, <code>IpduMContainerTxTriggerMode</code> is configured depending on the type of network that sends the <code>CONTAINER-I-PDU</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>IPDUM_DIRECT</code> for Ethernet and CAN</li> <li>▶ <code>IPDUM_TRIGGERTRANSMIT</code> for FlexRay and LIN</li> </ul> <p><code>IpduMContainerQueueSize</code> is set to <code>CONTAINER-I-PDU/MINIMUM-TX-CONTAINER-QUEUE-SIZE</code> or, as a fall-back, to <code>CONTAINER-I-PDU/SDGS/SDG [ @GID='IpduMContainerTxPduAttributes' ] /SD[ @GID='IpduMContainerQueueSize' ]</code> if the value is greater than the pre-existing <code>IpduMContainerQueueSize</code> value.</p>
<p><code>IpduMConfig/IpduM-ContainedRxPdu</code></p>	<p>For every PDU that is configured to be received within a <code>CONTAINER-I-PDU</code>, an <code>IpduMContainedRxPdu</code> container is created. A PDU is configured to be received within a <code>CONTAINER-I-PDU</code> if it meets the following conditions:</p> <ul style="list-style-type: none"> <li>▶ The type of the PDU is a subclass of <code>I-PDU</code>.</li> <li>▶ The PDU contains a <code>CONTAINED-I-PDU-PROPS</code> subelement.</li> <li>▶ The PDU is referenced by a <code>PDU-TRIGGERING</code>.</li> <li>▶ The imported <code>ECU-INSTANCE</code> owns an <code>I-PDU-PORT</code> that is referenced by the <code>PDU-TRIGGERING</code> and which has its <code>COMMUNICATION-DIRECTION</code> set to <code>IN</code>.</li> <li>▶ The <code>PDU-TRIGGERING</code> is referenced by at least one received <code>CONTAINER-I-PDU</code>.</li> </ul> <p>The name of the <code>IpduMContainedRxPdu</code> is set to <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the contained PDU.</p> <p><code>IpduMContainedPduHeaderId</code> is configured with either <code>CONTAINED-I-PDU-PROPS/HEADER-ID-LONG-HEADER</code> or <code>CONTAINED-I-PDU-PROPS/HEADER-ID-SHORT-HEADER</code> or not configured at all, depending on the <code>HEADER-TYPE</code> value of the <code>CONTAINER-I-PDU</code>:</p>

Configuration parameters	Mapping description
	<p>▶ CONTAINED-I-PDU-PROPS/HEADER-ID-LONG-HEADER is taken if HEADER-TYPE is set to LONG-HEADER.</p> <p>▶ CONTAINED-I-PDU-PROPS/HEADER-ID-SHORT-HEADER is taken if HEADER-TYPE is set to SHORT-HEADER.</p> <p>IpduMContainedPduHeaderId is not configured, if HEADER-TYPE is set to NO-HEADER.</p> <p>If the PDU is not associated with a CONTAINER-I-PDU or the associated CONTAINER-I-PDU does not provide a HEADER-TYPE, CONTAINED-I-PDU-PROPS/HEADER-ID-LONG-HEADER is taken. If the contained PDU does not provide CONTAINED-I-PDU-PROPS/HEADER-ID-LONG-HEADER, CONTAINED-I-PDU-PROPS/HEADER-ID-SHORT-HEADER is taken instead.</p> <p>IpduMContainedRxInContainerPduRef references the container created for the associated CONTAINER-I-PDU. The reference is configured if RX-ACCEPT-CONTAINED-I-PDU of the CONTAINER-I-PDU is set to a value other than ACCEPT-ALL or not set at all.</p> <p>IpduMContainedRxPduRef references the container created for the PDU in the EcuC module configuration.</p> <p>IpduMContainedPduOffset is set to CONTAINED-I-PDU-PROPS/OFFSET if the HEADER-TYPE is set to NO-HEADER.</p> <p>IpduMPduUpdateBitPosition is set to CONTAINED-I-PDU-PROPS/UPDATE-INDICATION-BIT-POSITION if the HEADER-TYPE is set to NO-HEADER.</p>
IpduMConfig/IpduM-ContainedTxPdu	<p>For every PDU that is sent within a CONTAINER-I-PDU, an IpduMContainedTxPdu container is created. A PDU is sent within a CONTAINER-I-PDU if it meets the conditions as follows:</p> <ul style="list-style-type: none"> <li>▶ The type of the PDU is a subclass of I-PDU.</li> <li>▶ The PDU contains a CONTAINED-I-PDU-PROPS subelement.</li> <li>▶ The PDU is referenced by a PDU-TRIGGERING.</li> <li>▶ The imported ECU-INSTANCE owns an I-PDU-PORT that is referenced by the PDU-TRIGGERING and which has its COMMUNICATION-DIRECTION set to OUT.</li> <li>▶ The PDU-TRIGGERING is referenced from the CONTAINER-I-PDU.</li> </ul>

Configuration parameters	Mapping description
	<p>The name of the <code>IpduMContainedTxPdu</code> is set to <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the contained PDU.</p> <p><code>IpduMContainedPduHeaderId</code> is configured in the same way as the corresponding parameter in <code>IpduMConfig/IpduMContainedRxPdu</code>.</p> <p><code>IpduMContainedTxPduCollectionSemantics</code> is configured depending on CONTAINED-I-PDU-PROPS/COLLECTION-SEMANTICS:</p> <ul style="list-style-type: none"> <li>▶ <code>IPDUM_COLLECT_LAST_IS_BEST</code> for LAST-IS-BEST</li> <li>▶ <code>IPDUM_COLLECT_QUEUED</code> for QUEUED</li> </ul> <p><code>IpduMContainedTxPduSendTimeout</code> is set to CONTAINED-I-PDU-PROPS/TIMEOUT.</p> <p><code>IpduMContainedTxPduTrigger</code> is configured depending on CONTAINED-I-PDU-PROPS/TRIGGER:</p> <ul style="list-style-type: none"> <li>▶ <code>IPDUM_TRIGGER_ALWAYS</code> for ALWAYS</li> <li>▶ <code>IPDUM_TRIGGER_NEVER</code> for NEVER</li> </ul> <p><code>IpduMContainedTxInContainerPduRef</code> references the container created for the associated CONTAINER-I-PDU.</p> <p><code>IpduMContainedTxPduRef</code> is configured in the same way as <code>IpduMContainedRxPduRef</code> in <code>IpduMConfig/IpduMContainedRxPdu</code>.</p> <p><code>IpduMContainedPduOffset</code> is configured in the same way as the corresponding parameter in <code>IpduMConfig/IpduMContainedRxPdu</code>.</p> <p><code>IpduMPduUpdateBitPosition</code> is configured in the same way as the corresponding parameter in <code>IpduMConfig/IpduMContainedRxPdu</code>.</p> <p><code>IpduMContainedTxPduPriority</code> is set to CONTAINED-I-PDU-PROPS/PRIORITY.</p>

### 3.4.30. J1939Dcm

Configuration parameters	Mapping description
J1939DcmConfigSet/ J1939DcmChannel	For each J-1939-CLUSTER the imported ECU-INSTANCE is attached to, one J1939DcmChannel container is created. Its name is J1939DcmChannel_<name> where <name> the short name of the J1939Cluster. J1939DcmComMChannelRef is set up to refer to the ComMChannel container that represents the J-1939-CLUSTER in the ComM BSW module configuration.
J1939DcmConfigSet/ J1939DcmNode	For each J-1939-CLUSTER the imported ECU-INSTANCE is attached to, one J1939DcmNode container is created. Its name is J1939DcmNode_<name> where <name> the short name of the J1939Cluster. If there is a J-1939-NM-NODE which refers to one of the COMMUNICATION-CONTROLLER elements of the imported ECU-INSTANCE and which is contained in a J-1939-NM-CLUSTER that refers to the J-1939-CLUSTER, then J1939DcmNmNodeRef is configured to refer to the container representing the J-1939-NM-NODE in the J1939Nm configuration.  J1939DcmNodeRmUserRef is configured to refer to the container in the J1939Rm BSW module that represents the J1939RM_USER_J1939DCM user of the J-1939-CLUSTER, as described in <a href="#">Section 3.4.32, "J1939Rm"</a> .
J1939RmConfigSet/ J1939DcmNode/ J1939DcmDiagnostic- MessageSupport	For each PDU of the J1939Rm user J1939Dcm_USER_J1939DCM, see <a href="#">Section 3.4.32, "J1939Rm"</a> , one J1939DcmDiagnosticMessageSupport container is created. Its name is the short name of the PDU.  J1939DcmDiagnosticMessageSupportChannelRef is set up to refer to the J1939DcmChannel container that represents the J-1939-CLUSTER in which the PDU is sent or received.  If the PDU has a PGN associated which the J1939Dcm configuration is supporting, i.e. [J1939DCM_DM01_SUPPORT..J1939DCM_DM57_SUPPORT], then J1939DcmDmxSupport is configured to that value.  If the imported ECU-INSTANCE sends the PDU, one J1939DcmTxPdu sub container is created. The container name is the short name of the PDU, J1939DcmTxPduRef is configured to refer to the EcuC container that has been created for the PDU.

### 3.4.31. J1939Nm

Configuration parameters	Mapping description
J1939NmConfigSet/ J1939NmChannel	<p>For each J-1939-NM-CLUSTER that contains a J-1939-NM-NODE which in turn refers via J-1939-NM-NODE/CONTROLLER-REF to a CAN-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE, one J1939NmChannel container is created. Its name is the short name of the J-1939-NM-CLUSTER. The following parameters are configured:</p> <p>If J-1939-NM-CLUSTER/COMMUNICATION-CLUSTER-REF refers to a valid J-1939-CLUSTER, J-1939-CLUSTER/USES-ADDRESS-ARBITRATION is used to configure J1939NmChannelUsesAddressArbitration.</p> <p>If J-1939-NM-CLUSTER/COMMUNICATION-CLUSTER-REF refers to a valid J-1939-CLUSTER, J1939NmComMNetworkHandleRef is set up to reference the ComMChannel container that represents this J-1939-CLUSTER in the ComM BSW module configuration.</p> <p>If the J-1939-NM-NODE that refers to a CAN-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE references NM-PDU elements via J-1939-NM-NODE/TX-NM-PDU-REFS and via J-1939-NM-NODE/RX-NM-PDU-REFS, and the imported ECU-INSTANCE actually sends/receives these PDUs, J1939NmTxPdu/J1939NmTxPduRef and J1939NmRxPdu/J1939NmRxPduRef are configured to reference the containers that represent these PDUs in the EcuC BSW module configuration.</p>
J1939NmConfigSet/ J1939NmNode	<p>For each J-1939-NM-NODE entity that refers via J-1939-NM-NODE/CONTROLLER-REF to a CAN-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE, one J1939NmNode container is created. Its name is the short name of the J-1939-NM-NODE entity. The following parameters are derived from the J-1939-NM-NODE and its NODE-NAME child entity:</p> <p>J1939NmNodeNameArbitraryAddressCapable is set to NODE-NAME/ARBITRARY-ADDRESS-CAPABLE.</p> <p>J1939NmNodeNameECUInstance is set to NODE-NAME/ECU-INSTANCE.</p> <p>J1939NmNodeNameFunction is set to NODE-NAME/FUNCTION.</p> <p>J1939NmNodeNameFunctionInstance is set to NODE-NAME/FUNCTION-INSTANCE.</p> <p>J1939NmNodeNameIdentityNumber is set to NODE-NAME/IDENTITY-NUMBER.</p>

Configuration parameters	Mapping description
	<p>J1939NmNodeNameIndustryGroup is set to NODE-NAME/INDUSTRY-GROUP.</p> <p>J1939NmNodeNameManufacturerCode is set to NODE-NAME/MANUFACTURER-CODE.</p> <p>J1939NmNodeNameVehicleSystem is set to NODE-NAME/VEHICLE-SYSTEM.</p> <p>J1939NmNodeNameVehicleSystemInstance is set to NODE-NAME/VEHICLE-SYSTEM-INSTANCE.</p> <p>J1939NmNodePreferredAddress is set to J-1939-NM-NODE/NM-NODE-ID.</p> <p>In J1939NmNodeChannelRef, one reference is configured which refers to the J1939NmChannel container that was created for the J-1939-NM-CLUSTER entity that contains the J-1939-NM-NODE.</p>

### 3.4.32. J1939Rm

Configuration parameters	Mapping description
J1939RmConfigSet/ J1939RmChannel	<p>For each J-1939-CLUSTER the imported Ecu Instance is attached to, one J1939RmChannel container is created. Its name is J1939RmChannel_&lt;name&gt; where &lt;name&gt; the short name of the J-1939-CLUSTER. For the configuration of its J1939RmRqstRxPdu, J1939RmRqstTxPdu, J1939RmAckmRxPdu, and J1939RmAckmTxPdu sub containers, every PDU that fulfills the following conditions is retrieved:</p> <ul style="list-style-type: none"> <li>▶ The PDU is sent or received by the imported ECU-INSTANCE</li> <li>▶ The PDU is directly contained in a CAN-FRAME</li> <li>▶ The CAN-FRAME-TRIGGERING of the CAN-FRAME has a valid CAN-FRAME-TRIGGERING/IDENTIFIER field assigned, CAN-FRAME-TRIGGERING/CAN-ADDRESSING-MODE is set to EXTENDED, and CAN-FRAME-TRIGGERING/J-1939-REQUESTABLE is either set to false or not defined.</li> <li>▶ The nine most significant bits of the <i>PGN</i> (Parameter Group Number) value, i.e. the bits [16..24] of CAN-FRAME-TRIGGERING/IDENTIFIER contain the value 0xEA, or 0xE8.</li> <li>▶ The PDU is not of type N-PDU</li> </ul>

Configuration parameters	Mapping description															
	<p>Depending on the send/receive direction of the collected PDUs, and the most significant bits of the <i>PGN</i> value, the following sub containers are configured along with a reference parameter that is set up to refer to the <i>EcuC</i> container representing the collected PDUs.</p> <table><tr><th>Direc- tion/PGN</th><th>Container</th><th>Reference Parameter</th></tr><tr><td>Tx/0xEA</td><td>J1939RmRqstTxPdu</td><td>J1939RmRqstTxPduRef</td></tr><tr><td>Rx/0xEA</td><td>J1939RmRqstRxPdu</td><td>J1939RmRqstRxPduRef</td></tr><tr><td>Tx/0xE8</td><td>J1939RmAckmTxPdu</td><td>J1939RmAckmTxPduRef</td></tr><tr><td>Rx/0xE8</td><td>J1939RmAckmRxPdu</td><td>J1939RmAckmRxPduRef</td></tr></table>	Direc- tion/PGN	Container	Reference Parameter	Tx/0xEA	J1939RmRqstTxPdu	J1939RmRqstTxPduRef	Rx/0xEA	J1939RmRqstRxPdu	J1939RmRqstRxPduRef	Tx/0xE8	J1939RmAckmTxPdu	J1939RmAckmTxPduRef	Rx/0xE8	J1939RmAckmRxPdu	J1939RmAckmRxPduRef
Direc- tion/PGN	Container	Reference Parameter														
Tx/0xEA	J1939RmRqstTxPdu	J1939RmRqstTxPduRef														
Rx/0xEA	J1939RmRqstRxPdu	J1939RmRqstRxPduRef														
Tx/0xE8	J1939RmAckmTxPdu	J1939RmAckmTxPduRef														
Rx/0xE8	J1939RmAckmRxPdu	J1939RmAckmRxPduRef														
J1939RmConfigSet/ J1939RmNode	<p>For each J-1939-CLUSTER the imported ECU-INSTANCE is attached to, one J1939RmNode container is created. Its name is J1939RmNode_&lt;name&gt; where &lt;name&gt; the short name of the J-1939-CLUSTER.</p> <p>One J1939RmNodeChannelRef container is configured to reference the corresponding J1939RmChannel container. If there is a J-1939-NM-NODE which refers to one of the COMMUNICATION-CONTROLLER elements of the imported ECU-INSTANCE and which is contained in a J-1939-NM-CLUSTER that refers to the J-1939-CLUSTER, then J1939RmNmNodeRef is configured to refer to the container representing the J-1939-NM-NODE in the J1939Nm configuration.</p> <p>For the configuration of the J1939RmUser sub containers, all PDUs of the J-1939-CLUSTER that can be requested by the J1939Rm BSW module are collected. They must satisfy the following conditions:</p> <ul style="list-style-type: none"><li>▶ The PDU is sent or received by the imported ECU-INSTANCE.</li><li>▶ The PDU is directly contained in a CAN-FRAME.</li><li>▶ The CAN-FRAME-TRIGGERING of the CAN-FRAME has a valid CAN-FRAME-TRIGGERING/IDENTIFIER field assigned, CAN-FRAME-TRIGGERING/CAN-ADDRESSING-MODE is set to EXTENDED, and CAN-FRAME-TRIGGERING/J-1939-REQUESTABLE is set to true.</li><li>▶ The PDU is not of type N-PDU.</li></ul> <p>Moreover, all PDUs of the J-1939-CLUSTER that are processed by the J1939Tp BSW module and which can be requested by the J1939Rm BSW module as well are collected. They fulfill the following conditions:</p> <ul style="list-style-type: none"><li>▶ The PDU is sent or received by the imported ECU-INSTANCE.</li></ul>															



Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ There is a J-1939-TP-CONNECTION which contains a J-1939-TP-PG element which in turn references the PDU via J-1939-TP-PG/SDU-REFS/SDU-REF.</li> <li>▶ The J-1939-TP-CONNECTION is either sent or received by the imported Ecu Instance and it belongs to a J-1939-TP-CONFIG that references the J-1939-CLUSTER.</li> <li>▶ The J-1939-TP-PG has its J-1939-TP-PG/REQUESTABLE field set to true.</li> </ul> <p>The two collections of PDUs are split into subgroups each of which belongs to a given J1939RmUser.</p> <ul style="list-style-type: none"> <li>▶ The PDUs for J1939RM_USER_COM are of type I-SIGNAL-I-PDU.</li> <li>▶ The PDUs for J1939RM_USER_J1939NM are of type NM-PDU, they are sent directly in a CAN-FRAME, and they must have assigned the PGN 0xee00.</li> <li>▶ The PDUs for J1939RM_USER_J1939DCM are of type J-1939-DCM-I-PDU.</li> </ul>
J1939RmConfigSet/ J1939RmNode/ J1939RmUser/ J1939RmUser_Com	<p>If PDUs for J1939RM_USER_COM exist, one J1939RmUser_Com container is created, J1939RmUserType is set to J1939RM_USER_COM. For each of the PDUs, one J1939RmComIPdu sub container is created, its name is the short name of the PDU. ComIPduType is set to ComIPdu_TP for PDUs that either have a dynamic length or a length that exceeds eight bytes. For all other PDUs, ComIPduType is set to ComIPdu_IF. An I-SIGNAL-I-PDU has a dynamic length if at least one of its I-SIGNAL elements refers to a SYSTEM-SIGNAL that has its DYNAMIC-LENGTH field set to true.</p> <p>For PDUs processed by J1939Tp, the parameters J1939RmComIPduDA, J1939RmComIPduSA are configured according to the J1939TpTxDa/J1939TpRxSa and J1939TpTxSa/J1939TpRxSa parameters of the J1939TpTxChannel or J1939TpRxChannel the PDU is processed in, see <a href="#">Section 3.4.33, "J1939Tp"</a>. J1939RmComIPduPGN is configured in the same way as the J1939TpTxPgPGN/J1939TpRxPgPGN parameter values of the J1939TpTxPg/J1939TpRxPg containers that were set up for the PDUs in the J1939Tp module configuration.</p> <p>For PDUs that are directly sent/received in a CAN-FRAME, the J1939RmComIPduDA, J1939RmComIPduSA, and J1939RmComIPduPGN are extracted from CAN-FRAME-TRIGGERING/IDENTIFIER:</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ If the bit range [16..23] yields a value that is less than 240, J1939RmComIPduDA is set to zero. Otherwise J1939RmComIPduDA is set to the value of the bit range [8..15].</li> <li>▶ J1939RmComIPduSA is set to the value of the bit range [0..7].</li> <li>▶ If the bit range [16..23] yields a value that is less than 240, J1939RmComIPduPGN is set to the value of the bit range [16..24] multiplied by 256. Otherwise J1939RmComIPduPGN is set to the value of the bit range [8..24].</li> </ul> <p>One J1939RmLowerComIPdu and one J1939RmUpperComIPdu container are created, their name is the short name of the PDU, prefixed by LO for the J1939RmLowerComIPdu and prefixed by UP for the J1939RmUpperComIPdu container. The reference parameters J1939RmLowerComIPduId/J1939RmLowerComIPduRef and J1939RmUpperComIPdu/J1939RmUpperComIPduRef are configured to refer to the representation of the PDU in the EcuC module.</p>
J1939RmConfigSet/ J1939RmNode/ J1939RmUser/ J1939RmUser_J1939Nm	If PDUs for J1939RM_USER_J1939NM exist, one J1939RmUser_J1939Nm container is created, J1939RmUserType is set to J1939RM_USER_J1939NM. One J1939RmUserPGN entry is created, its value is set to 0xee00.
J1939RmConfigSet/ J1939RmNode/ J1939RmUser/ J1939RmUser_J1939Dcm	If PDUs for J1939RM_USER_J1939DCM exist, one J1939RmUser_J1939Dcm container is created, J1939RmUserType is set to J1939RM_USER_J1939DCM. For every PDU, its PGN is determined in the same way as for the J1939RmComIPduPGN value of J1939RM_USER_COM PDUs. For each valid PGN value, one J1939RmUserPGN entry is created, its value is set to the PGN value.

### 3.4.33. J1939Tp

Configuration parameters	Mapping description
J1939TpConfiguration/ J1939TpTxChannel, J1939TpConfiguration/ J1939TpRxChannel	<p>For each J-1939-TP-CONNECTION which the imported Ecu Instance either receives or sends, and which belongs to a J-1939-TP-CONFIG that references a J-1939-CLUSTER, one J1939TpRxChannel or one J1939TpTxChannel is created.</p> <p>The imported ECU-INSTANCE receives a J-1939-TP-CONNECTION if it references a J-1939-TP-NODE via J-1939-TP-CONNECTION/RECEIVER-REF and if the J-1939-TP-NODE refers via J-1939-TP-NODE/</p>

Configuration parameters	Mapping description
	<p>CONNECTOR-REF to a CAN-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE.</p> <p>The imported ECU-INSTANCE sends a J-1939-TP-CONNECTION if it references a J-1939-TP-NODE via J-1939-TP-CONNECTION/TRANSMITTER-REF and if the J-1939-TP-NODE refers via J-1939-TP-NODE/CONNECTOR-REF to a CAN-COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE.</p> <p>The container name is the short name of the J-1939-TP-CONNECTION. The following parameters are set for J1939TpTxChannel containers:</p> <p>J1939TpTxCancellationSupport is set to J-1939-TP-CONNECTION/CANCELLATION.</p> <p>If J-1939-TP-CONNECTION/RECEIVER-REF refers to a valid J-1939-TP-NODE, and J-1939-TP-NODE/TP-ADDRESS-REF refers to a valid TP-ADDRESS, TP-ADDRESS/TP-ADDRESS is used to configure J1939TpTxDa.</p> <p>J1939TpTxDynamicBlockCalculation is set to J-1939-TP-CONNECTION/DYNAMIC-BS.</p> <p>J1939TpTxMaxPacketsPerBlock is set to J-1939-TP-CONNECTION/MAX-EXP-BS.</p> <p>J1939TpTxProtocolType is set to J1939TP_PROTOCOL_BAM if J-1939-TP-CONNECTION/BROADCAST is true, it is set to J1939TP_PROTOCOL_CMDT if J-1939-TP-CONNECTION/BROADCAST is false</p> <p>J1939TpTxRetrySupport is set to J-1939-TP-CONNECTION/RETRY.</p> <p>If J-1939-TP-CONNECTION/TRANSMITTER-REF refers to a valid J-1939-TP-NODE, and J-1939-TP-NODE/TP-ADDRESS-REF refers to a valid TP-ADDRESS, TP-ADDRESS/TP-ADDRESS is used to configure J1939TpTxSa.</p> <p>The following parameters are set for J1939TpRxChannel containers:</p> <p>J1939TpRxCancellationSupport is configured in the same way as J1939TpTxCancellationSupport.</p> <p>J1939TpRxDa is configured in the same way as J1939TpTxDa.</p> <p>J1939TpRxDynamicBlockCalculation is configured in the same way as J1939TpTxDynamicBlockCalculation.</p>

Configuration parameters	Mapping description
	<p>J1939TpRxDynamicBufferRatio is set to J-1939-TP-CONNECTION/BUFFER-RATIO.</p> <p>J1939TpRxPacketsPerBlock is set to J-1939-TP-CONNECTION/MAX-BS.</p> <p>J1939TpRxProtocolType is configured in the same way as J1939TpTxProtocolType.</p> <p>J1939TpRxRetrySupport is configured in the same way as J1939TpTxRetrySupport.</p> <p>J1939TpRxSa is configured in the same way as J1939TpTxSa.</p>
N-PDU containers of J1939TpTxChannels	<p>If the imported ECU-INSTANCE sends one of the PDUs that are referenced by J-1939-TP-CONNECTION/FLOW-CONTROL-PDU-REFS, one J1939TpTxCmNPdu container is created, its name is the short name of the PDU. J1939TpTxCmNPduRef is configured to reference the representation of the PDU in the EcuC module.</p> <p>If the imported ECU-INSTANCE receives one of the PDUs that are referenced by J-1939-TP-CONNECTION/FLOW-CONTROL-PDU-REFS, one J1939TpRxFcNPdu container is created, its name is the short name of the PDU. J1939TpRxFcNPduRef is configured to reference the representation of the PDU in the EcuC module.</p> <p>If the imported ECU-INSTANCE sends the PDU that is referenced by J-1939-TP-CONNECTION/DATA-PDU-REF, one J1939TpTxDtNPdu container is created, its name is the short name of the PDU. J1939TpTxDtNPduRef is configured to reference the representation of the PDU in the EcuC module.</p>
N-PDU containers of J1939TpRxChannels	<p>If the imported ECU-INSTANCE receives one of the PDUs that are referenced by J-1939-TP-CONNECTION/FLOW-CONTROL-PDU-REFS, one J1939TpRxCmNPdu container is created, its name is the short name of the PDU. J1939TpRxCmNPduRef is configured to reference the representation of the PDU in the EcuC module.</p> <p>If the imported ECU-INSTANCE sends one of the PDUs that are referenced by J-1939-TP-CONNECTION/FLOW-CONTROL-PDU-REFS, one J1939TpTxFcNPdu container is created, its name is the short name of the PDU. J1939TpTxFcNPduRef is configured to reference the representation of the PDU in the EcuC module.</p>

Configuration parameters	Mapping description
	<p>If the imported ECU-INSTANCE receives the PDU that is referenced by J-1939-TP-CONNECTION/DATA-PDU-REF, one J1939TpRxDtNPdu container is created, its name is the short name of the PDU.</p> <p>J1939TpRxDtNPduRef is configured to reference the representation of the PDU in the EcuC module.</p>
J1939TpTxChannel/ J1939TpTxPg	<p>For every J-1939-TP-PG of the J-1939-TP-CONNECTION which references a PDU which the imported Ecu instance is sending via J-1939-TP-PG/SDU-REFS or J-1939-TP-PG/DIRECT-PDU-REF, one J1939TpTxPg container is created. Its name is either Pg_&lt;pgn&gt; if J-1939-TP-PG/PGN is available, or Pg_Idx_&lt;idx&gt; if J-1939-TP-PG/PGN is not available, where &lt;idx&gt; is a zero based index. The following parameters are configured:</p> <p>If there is at least one PDU that is referenced via J-1939-TP-PG/SDU-REFS and that the imported Ecu Instance sends is an I-SIGNAL-I-PDU containing at least one I-SIGNAL, J1939TpTxPgDynLength is configured. The parameter is set to true if at least one of the I-SIGNAL elements refers to a SYSTEM-SIGNAL that has its DYNAMIC-LENGTH field set to true. Otherwise the parameter is set to false.</p> <p>J1939TpTxPgPGN is set to J-1939-TP-PG/PGN.</p> <p>If the imported ECU-INSTANCE sends the PDU that is referenced by J-1939-TP-PG/DIRECT-PDU-REF, one J1939TpTxDirectNPdu container is created, its name is the short name of the PDU.</p> <p>J1939TpTxDirectNPduRef is configured to reference the representation of the PDU in the EcuC module.</p> <p>For each PDU that is referenced via J-1939-TP-PG/SDU-REFS and which the imported ECU-INSTANCE is sending, one J1939TpTxNSdu container is created, its name is the short name of the PDU. J1939TpTxNSduRef is configured to reference the representation of the PDU in the EcuC module.</p>
J1939TpRxChannel/ J1939TpRxPg	<p>For every J-1939-TP-PG of the J-1939-TP-CONNECTION which references a PDU which the imported Ecu instance is receiving via J-1939-TP-PG/SDU-REFS or J-1939-TP-PG/DIRECT-PDU-REF, one J1939TpRxPg container is created. Its name is created in the same way as it is for J1939TpTxPg containers. The following parameters are configured:</p> <p>J1939TpRxPgDynLength is configured in the same way as J1939TpTxPgDynLength.</p> <p>J1939TpRxPgPGN is configured in the same way as J1939TpTxPgPGN.</p>

Configuration parameters	Mapping description
	<p>If the imported ECU-INSTANCE receives the PDU that is referenced by J-1939-TP-PG/DIRECT-PDU-REF, one J1939TpRxDirectNPdu container is created, its name is the short name of the PDU. J1939TpRxDirectNPduRef is configured to reference the representation of the PDU in the EcuC module.</p> <p>For each PDU that is referenced via J-1939-TP-PG/SDU-REFS and which the imported ECU-INSTANCE is receiving, one J1939TpRxNSdu container is created, its name is the short name of the PDU. J1939TpRxNSduRef is configured to reference the representation of the PDU in the EcuC module.</p>

### 3.4.34. LdCom

Configuration parameters	Mapping description
LdComIPdu	<p>For every PDU instance (see <a href="#">Section 3.3.1.3, "Instance handling"</a>) that represents an I-SIGNAL-I-PDU, which is sent or received by the imported ECU-INSTANCE and for which the following conditions apply, an LdComIPdu container is created.</p> <ul style="list-style-type: none"> <li>▶ The PDU instance contains exactly one I-SIGNAL-TO-I-PDU-MAPPING referencing an I-SIGNAL.</li> <li>▶ ComSignalType of the I-SIGNAL is UINT8_N or UINT8_DYN.</li> <li>▶ PACKING-BYTE-ORDER of the I-SIGNAL-TO-I-PDU-MAPPING is set to OPAQUE.</li> <li>▶ START-POSITION of the I-SIGNAL-TO-I-PDU-MAPPING is set to 0.</li> <li>▶ UPDATE-INDICATION-BIT-POSITION of the I-SIGNAL-TO-I-PDU-MAPPING is not set.</li> <li>▶ The I-SIGNAL-PORT referenced by the I-SIGNAL's I-SIGNAL-TRIGGERING has no TIMEOUT defined.</li> <li>▶ The I-SIGNAL-PORT referenced by the I-SIGNAL's I-SIGNAL-TRIGGERING has no FIRST-TIMEOUT defined.</li> <li>▶ The I-SIGNAL-PORT referenced by the I-SIGNAL's I-SIGNAL-TRIGGERING either has no DATA-FILTER defined or has DATA-FILTER/DATA-FILTER-TYPE set to ALWAYS.</li> <li>▶ There is no I-SIGNAL-I-PDU-GROUP which references the I-SIGNAL-I-PDU.</li> </ul>

Configuration parameters	Mapping description
	<p>► There is no I-SIGNAL-MAPPING which references the I-SIGNAL's I-SIGNAL-TRIGGERING.</p> <p>The following conditions only apply for I-SIGNAL-I-PDUS, whose LdComIPduDirection is set to LDCOM_SEND.</p> <ul style="list-style-type: none"> <li>► I-PDU-TIMING-SPECIFICATIONS/I-PDU-TIMING/TRANSMISSION-MODE-DECLARATION is defined.</li> <li>► TRANSFER-PROPERTY of the I-SIGNAL-TO-I-PDU-MAPPING is set to TRIGGERED or TRIGGERED-WITHOUT-REPETITION.</li> <li>► MINIMUM-DELAY of the I-PDU-TIMING is not set if the LdComIPduDirection is set to LDCOM_SEND.</li> <li>► The I-PDU-TIMING has a TRANSMISSION-MODE-TRUE-TIMING but no TRANSMISSION-MODE-FALSE-TIMING defined.</li> <li>► The TRANSMISSION-MODE-DECLARATION either has no TRANSMISSION-MODE-CONDITIONS defined or one single TRANSMISSION-MODE-CONDITION which references the I-SIGNAL-TO-I-PDU-MAPPING and which has its DATA-FILTER/DATA-FILTER-TYPE set to ALWAYS.</li> <li>► The TRANSMISSION-MODE-DECLARATION has no MODE-DRIVEN-FALSE-CONDITIONS and no MODE-DRIVEN-TRUE-CONDITIONS defined.</li> <li>► The TRANSMISSION-MODE-TRUE-TIMING has only an EVENT-CONTROLLED-TIMING defined and its NUMBER-OF-REPETITIONS must be set to 0.</li> </ul> <p>The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the I-SIGNAL-I-PDU.</p> <p>LdComPduRef references the corresponding container in the EcuC module configuration.</p> <p>LdComSystemTemplateSignalRef is set to the AUTOSAR path of the I-SIGNAL-TO-I-PDU-MAPPING.</p> <p>LdComApiType is configured in the same way as ComIPduType in <a href="#">Section 3.4.8, "Com"</a>, with the exception that LDCOM_TP and LDCOM_IF are used as configured parameter values instead of TP and NORMAL.</p> <p>LdComIPduDirection is set to LDCOM_SEND if the PDU instance is sent by the imported ECU-INSTANCE. Otherwise it is set to LDCOM_RECEIVE.</p>

### 3.4.35. Lin

Configuration parameters	Mapping description
LinChannel	<p>For every LIN-CLUSTER which belongs to the imported ECU-INSTANCE, a LinChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the LIN-CLUSTER's PHYSICAL-CHANNEL. A LIN-CLUSTER belongs to the imported ECU-INSTANCE if a LIN-MASTER connects the LIN-CLUSTER to the imported ECU-INSTANCE via its COMMUNICATION-CONNECTOR.</p> <p>If a LIN-CLUSTER is not connected to the imported ECU-INSTANCE by a LIN-MASTER, a warning is reported and no LinChannel container is created for this LIN-CLUSTER.</p> <p>If the LIN-CLUSTER comprises more than one PHYSICAL-CHANNEL, an error is reported.</p> <p>LinChannelBaudRate is set to BAUDRATE. If BAUDRATE is not available, LinChannelBaudRate is set to SPEED * 1000.</p>

### 3.4.36. LinIf

Configuration parameters	Mapping description
LinIfChannel	<p>For every LIN-CLUSTER which belongs to the imported ECU-INSTANCE, a LinIfChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the LIN-CLUSTER's PHYSICAL-CHANNEL. A LIN-CLUSTER belongs to the imported ECU-INSTANCE if a LIN-MASTER connects the LIN-CLUSTER to the imported ECU-INSTANCE via its COMMUNICATION-CONNECTOR.</p> <p>If a LIN-CLUSTER is not connected to the imported ECU-INSTANCE by a LIN-MASTER, a warning is reported and no LinIfChannel container is created for this LIN-CLUSTER.</p> <p>If the LIN-CLUSTER comprises more than one PHYSICAL-CHANNEL, an error is reported.</p> <p>If the single PHYSICAL-CHANNEL is not a LIN-PHYSICAL-CHANNEL, an error is reported.</p>



Configuration parameters	Mapping description
	<p><code>LinIfChannelRef</code> references the corresponding container in the <code>Lin</code> module configuration.</p> <p><code>LinIfComMNetworkHandleRef</code> references the corresponding container in the <code>ComM</code> module configuration.</p>
<code>LinIfChannel/LinIf-Master</code>	<p>The following parameters are set using the <code>LIN-MASTER</code> connecting the imported ECU-INSTANCE to the <code>LIN-CLUSTER</code>:</p> <p><code>LinIfClusterTimeBase</code> is set to <code>TIME-BASE</code>.</p> <p><code>LinIfJitter</code> is set to <code>TIME-BASE-JITTER</code>.</p>
<code>LinIfChannel/LinIfSlave</code>	<p>If the <code>LIN-MASTER</code> of a given <code>LIN-CLUSTER</code> contains one or more <code>LIN-SLAVE-CONFIG</code> elements, these are used to configure the <code>LinIfSlave</code> configuration containers. If the <code>LIN-MASTER</code> does not contain any <code>LIN-SLAVE-CONFIG</code>, all <code>LIN-SLAVE</code> elements that are connected to the <code>LIN-CLUSTER</code> are used to configure the <code>LinIfSlave</code> configuration containers.</p> <p>For every <code>LIN-SLAVE-CONFIG</code> or <code>LIN-SLAVE</code> element one <code>LinIfSlave</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of either <code>LIN-SLAVE-CONFIG/IDENT</code> or <code>LIN-SLAVE</code>.</p> <p><code>LinIfProtocolVersion</code> is set to <code>PROTOCOL-VERSION</code>.</p> <p><code>LinIfConfiguredNad</code> is set to <code>CONFIGURED-NAD</code>.</p> <p><code>LinIfFunctionId</code> is set to <code>FUNCTION-ID</code>.</p> <p><code>LinIfSupplierId</code> is set to <code>SUPPLIER-ID</code>.</p> <p><code>LinIfVariant</code> is set to <code>VARIANT-ID</code>.</p>
<code>LinIfChannel/LinIf-Frame</code>	<p>In the following, <code>APPLICATION-ENTRY</code> elements referencing <code>FRAME-TRIGGERING</code> elements are referred to as <code>ApplicationFrame</code> elements; all other types of <code>ENTRY</code> elements do not reference <code>FRAME-TRIGGERING</code> elements and are referred to as <code>ConfigurationFrame</code> elements. <code>SlaveToSlaveFrame</code> elements are a special kind of <code>ApplicationFrame</code> elements which are neither sent nor received by the <code>LIN-MASTER</code>.</p> <p>For every kind of frame, a <code>LinIfFrame</code> container is created. The container name for <code>ApplicationFrame</code> elements is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, for <code>SlaveToSlaveFrame</code> elements it is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>FRAME</code> referenced by the <code>LIN-FRAME-TRIGGERING</code>. The container name for <code>ConfigurationFrame</code> elements is</p>

Configuration parameters	Mapping description
	<p>&lt;PREFIX&gt;&lt;name&gt;&lt;auto incremented number&gt;, where &lt;name&gt; is the SHORT-NAME of the parent LIN-SCHEDULE-TABLE.</p> <p>LinIfPid is set to the protected identifier of the LIN-FRAME-TRIGGERING. For details on how to calculate the protected identifier refer to [2]. If the LinIfFrame has been created for a ConfigurationFrame, LinIfPid is set to 60, which is the ID of the MasterRequestFrame (MRF). If the LIN-FRAME-TRIGGERING references a LIN-SPORADIC-FRAME, LinIfPid is not configured.</p> <p>LinIfLength is set to FRAME-LENGTH of the LIN-UNCONDITIONAL-FRAME referenced by the LIN-FRAME-TRIGGERING. If the LIN-FRAME-TRIGGERING references a LIN-SPORADIC-FRAME or LIN-EVENT-TRIGGERED-FRAME, the FRAME-LENGTH of the first LIN-UNCONDITIONAL-FRAME referenced by the LIN-SPORADIC-FRAME or LIN-EVENT-TRIGGERED-FRAME is used. For ConfigurationFrame elements LinIfLength is always set to 8.</p> <p>LinIfFrameType is set depending on the IDENTIFIER of the LIN-FRAME-TRIGGERING:</p> <ul style="list-style-type: none"> <li>▶ MRF for 60</li> <li>▶ SRF for 61</li> </ul> <p>For other values of IDENTIFIER LinIfFrameType is set depending on the type of the referenced FRAME:</p> <ul style="list-style-type: none"> <li>▶ UNCONDITIONAL for LIN-UNCONDITIONAL-FRAME elements</li> <li>▶ SPORADIC for LIN-SPORADIC-FRAME elements</li> <li>▶ EVENT_TRIGGERED for LIN-EVENT-TRIGGERED-FRAME elements</li> </ul> <p>For ConfigurationFrame elements LinIfFrameType is set depending on the type of the TABLE-ENTRY:</p> <ul style="list-style-type: none"> <li>▶ FREE for FREE-FORMAT-ENTRY and DATA-DUMP-ENTRY</li> <li>▶ SAVE_CONFIGURATION for SAVE-CONFIGURATION-ENTRY</li> <li>▶ ASSIGN_NAD for ASSIGN-NAD</li> <li>▶ ASSIGN for ASSIGN-FRAME-ID</li> <li>▶ UNASSIGN for UNASSIGN-FRAME-ID</li> <li>▶ ASSIGN_FRAME_ID_RANGE for ASSIGN-FRAME-ID-RANGE</li> <li>▶ CONDITIONAL for CONDITIONAL-CHANGE-NAD</li> </ul>

Configuration parameters	Mapping description
	<p>For ConfigurationFrame elements LinIfChecksumType is set to CLASSIC. For LIN-FRAME-TRIGGERING elements referencing a LIN-UNCONDITIONAL-FRAME, LinIfChecksumType is set to LIN-CHECKSUM.</p> <p>For LIN-SPORADIC-FRAME elements and LIN-EVENT-TRIGGERED-FRAME elements LIN-CHECKSUM is taken from the LIN-FRAME-TRIGGERING elements referencing the substituted LIN-UNCONDITIONAL-FRAME elements. If inconsistencies are detected among the LIN-CHECKSUM values of these LIN-FRAME-TRIGGERING elements, a warning is reported.</p>
LinIfChannel/LinIfFixedFrame/LinIfFixedFrameSduByte	<p>For ConfigurationFrame elements the byte array used to configure LinIfFixedFrameSduByte is calculated depending on the type of the TABLE-ENTRY. In the following, array elements are listed starting from byte zero, and the parameters CONFIGURED-NAD, SUPPLIER-ID, and FUNCTION-ID are either taken from the LIN-SLAVE-CONFIG referenced via ASSIGNED-LIN-SLAVE-CONFIG-REF or, if this reference does not exist, from the LIN-SLAVE referenced via ASSIGNED-CONTROLLER-REF.</p> <ul style="list-style-type: none"> <li>▶ FREE-FORMAT: the byte array is taken directly from BYTE-VALUES.</li> <li>▶ ASSIGN-NAD: INITIAL-NAD of LIN-COMMUNICATION-CONNECTOR or INITIAL-NAD of LIN-SLAVE-CONFIG, 0x06, 0xb0, LSB of SUPPLIER-ID, MSB of SUPPLIER-ID, LSB of MESSAGE-ID, MSB of MESSAGE-ID, NEW-NAD .</li> <li>▶ SAVE-CONFIGURATION-ENTRY: CONFIGURED-NAD, 0x01, 0xb6, 0xff, 0xff, 0xff, 0xff .</li> <li>▶ ASSIGN-FRAME-ID/UNASSIGN-FRAME-ID : CONFIGURED-NAD, 0x06, 0xb1, LSB of SUPPLIER-ID, MSB of SUPPLIER-ID, LSB of MESSAGE-ID, MSB of MESSAGE-ID, protected identifier of LIN-FRAME-TRIGGERING.</li> </ul> <p>The LIN-FRAME-TRIGGERING is referenced by ASSIGN-FRAME-ID/UNASSIGN-FRAME-ID via ASSIGNED-FRAME-TRIGGERING-REF/UNASSIGNED-FRAME-TRIGGERING-REF.</p> <p>If the ASSIGN-FRAME-ID/UNASSIGN-FRAME-ID entity references a LIN-SLAVE-CONFIG entity, the MESSAGE-ID is directly taken from ASSIGN-FRAME-ID/MESSAGE-ID, respectively from UNASSIGN-FRAME-ID/MESSAGE-ID.</p> <p>If the ASSIGN-FRAME-ID/UNASSIGN-FRAME-ID entity references a LIN-SLAVE entity, the MESSAGE-ID is taken from the LIN-CON-</p>

Configuration parameters	Mapping description
	<p>FIGURABLE-FRAME referencing the same LIN-FRAME as the LIN-FRAME-TRIGGERING.</p> <p>► ASSIGN-FRAME-ID-RANGE: CONFIGURED-NAD, 0x06, 0xb7, START-INDEX, FRAME-PID[INDEX = 0]/PID, FRAME-PID[INDEX = 1]/PID, FRAME-PID[INDEX = 2]/PID, FRAME-PID[INDEX = 3]/PID.</p> <p>If ASSIGN-FRAME-ID-RANGE does not contain any FRAME-PID elements and the ASSIGN-FRAME-ID-RANGE refers to a LIN-SLAVE entity, the PID values are either retrieved from the LIN-ORDERED-CONFIGURABLE-FRAMES or, if these are not available, from the LIN-CONFIGURABLE-FRAMES of the LIN-SLAVE that is referenced via ASSIGNED-CONTROLLER-REF. The LIN-ORDERED-CONFIGURABLE-FRAMES and LIN-CONFIGURABLE-FRAMES of a slave are contained in the LIN-COMMUNICATION-CONNECTOR belonging to the LIN-SLAVE.</p> <p>The index of a CONFIGURABLE-FRAME element in LIN-ORDERED-CONFIGURABLE-FRAMES is determined by the INDEX parameter of the element. The index of a CONFIGURABLE-FRAME element in LIN-CONFIGURABLE-FRAMES is determined by its position, starting with zero as the index of the first element.</p> <p>Each of the CONFIGURABLE-FRAME elements in the list owns an IDENTIFIER, which is stored in the FRAME-TRIGGERING referencing the same FRAME that is referenced by the CONFIGURABLE-FRAME. The PID of a CONFIGURABLE-FRAME is its IDENTIFIER plus two parity bits. For details on the parity bits refer to <a href="#">[2]</a>.</p> <p>The bytes [4 .. 7] are filled with the PID of the CONFIGURABLE-FRAME at index [START-INDEX .. START-INDEX + 3].</p> <p>If no valid FRAME-PID or CONFIGURABLE-FRAME is found at a given INDEX, 0xff is configured.</p> <p>► CONDITIONAL-CHANGE-NAD: CONFIGURED-NAD, 0x06, 0xb3, ID, BYTE, MASK, INVERT, NEW-NAD.</p> <p>► DATA-DUMP-ENTRY: CONFIGURED-NAD, 0x06, 0xb4, BYTE-VALUES/BYTE-VALUE[0], BYTE-VALUES/BYTE-VALUE[1], BYTE-VALUES/BYTE-VALUE[2], BYTE-VALUES/BYTE-VALUE[3], BYTE-VALUES/BYTE-VALUE[4].</p> <p>If the number of BYTE-VALUE elements is not exactly five, an error is reported.</p>

Configuration parameters	Mapping description
	<p>If any of the values required to fill the byte array cannot be retrieved, an error is reported.</p> <p>For each byte of the array, a <code>LinIfFixedFrameSduByte</code> is created. The container name is <code>LinIfFixedFrameSduByte_&lt;byte position within array&gt;</code>.</p> <p><code>LinIfFixedFrameSduBytePos</code> is set to the byte position within the array.</p> <p><code>LinIfFixedFrameSduByteVal</code> is set to <code>&lt;byte value&gt;</code>, where only the eight least significant bits are considered.</p>
<code>LinIfChannel/LinIfFrame/LinIfPduDirection</code>	<p>For LIN-FRAME-TRIGGERING elements referencing a LIN-SPORADIC-FRAME and which have their IDENTIFIER set to values other than 60 or 61, the choice container is set to <code>LinIfRxPdu</code> (received by LIN-MASTER) or <code>LinIfTxPdu</code> (sent by LIN-MASTER). The name of <code>LinIfRxPdu/LinIfTxPdu</code> is set to <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the PDU belonging to the PDU instance (see <a href="#">Section 3.3.1.3, "Instance handling"</a>) of the LIN-FRAME-TRIGGERING.</p> <p><code>LinIfRxPduRef/LinIfTxPduRef</code> references the container in the <code>EcuC</code> module configuration, which has been created for the PDU instance of the LIN-FRAME-TRIGGERING. If the number of PDU instances for the LIN-FRAME-TRIGGERING is not exactly one, no properties are exported for <code>LinIfRxPdu/LinIfTxPdu</code> and a warning is reported.</p> <p><code>LinIfUserRxIndicationUL/LinIfUserTxUL</code> is set to PDUR for I-SIGNAL-PDU elements, MULTIPLEXED-I-PDU elements, CONTAINER-I-PDU elements, GENERAL-PURPOSE-I-PDU elements, DCM-I-PDU elements, and for USER-DEFINED-I-PDU elements, which either have a CATEGORY other than XCP or are routed (see <a href="#">Section 3.3.6, "PDU routing"</a>), or to CDD for received or sent USER-DEFINED-I-PDU elements that have their CATEGORY set to XCP and for all other PDU types.</p> <p>For LIN-FRAME-TRIGGERING elements not sent/received by the LIN-MASTER, but referenced by a TABLE-ENTRY, the <code>LinIfPduDirection</code> choice container is set to <code>LinIfSlaveToSlavePdu</code>. For all other TABLE-ENTRY types, the choice container is set to <code>LinIfInternalPdu</code>.</p>
<code>LinIfChannel/LinIfFrame/LinIfSubstitutionFrame</code>	<p>For every LIN-UNCONDITIONAL-FRAME referenced by a LIN-SPORADIC-FRAME via SUBSTITUTED-FRAME-REF, or by a LIN-EVENT-TRIGGERED-FRAME via LIN-UNCONDITIONAL-FRAME-REF, a <code>LinIfSubsti-</code></p>

Configuration parameters	Mapping description
	<p>tutionFrame container is created. The container name is the same as the LinIfFrame created for the referenced LIN-UNCONDITIONAL-FRAME.</p> <p>LinIfSubstitutionFrameRef references the container created for the referenced LIN-UNCONDITIONAL-FRAME.</p> <p>LinIfFramePriority is set to the position of the referring element (SUBSTITUTED-FRAME-REF/LIN-UNCONDITIONAL-FRAME-REF) within its parent element.</p>
LinIfChannel/LinIfScheduleTable	<p>For each LIN-SCHEDULE-TABLE a LinIfScheduleTable container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the LIN-SCHEDULE-TABLE.</p> <p>LinIfScheduleTableName is set to SHORT-NAME.</p> <p>LinIfRunMode is set depending on RUN-MODE:</p> <ul style="list-style-type: none"> <li>▶ RUN_CONTINUOUS for RUN-CONTINUOUS</li> <li>▶ RUN_ONCE for RUN-ONCE</li> </ul> <p>LinIfResumePosition is set depending on RESUME-POSITION:</p> <ul style="list-style-type: none"> <li>▶ CONTINUE_AT_IT_POINT for CONTINUE-AT-IT-POSITION</li> <li>▶ START_FROM_BEGINNING for START-FROM-BEGINNING</li> </ul>
LinIfChannel/LinIfScheduleTable/LinIfEntry	<p>For each TABLE-ENTRY a LinIfEntry container is created. The container name is Entry&lt;auto incremented number&gt;.</p> <p>LinIfDelay is set to DELAY.</p> <p>LinIfEntryIndex is set to POSITION-IN-TABLE.</p> <p>LinIfFrameRef references the corresponding LinIfFrame container.</p> <p>LinIfCollisionResolvingRef is only set for TABLE-ENTRY elements which reference a LIN-FRAME-TRIGGERING referencing a LIN-EVENT-TRIGGERED-FRAME. It references the LinIfScheduleTable created for the LIN-SCHEDULE-TABLE referenced via COLLISION-RESOLVING-SCHEDULE-REF.</p>

### 3.4.37. LinSM

Configuration parameters	Mapping description
LinSMCon- figSet/LinSMChannel	<p>For every LIN-CLUSTER which belongs to the imported ECU-INSTANCE, a LinSMChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the LIN-CLUSTER.</p> <p>LinSMComMNetworkHandleRef references the ComMChannel container in the ComM module configuration, which has been created for the LIN-CLUSTER.</p>
LinSMCon- figSet/LinSMChan- nel/LinSMSchedule	<p>For every LIN-SCHEDULE-TABLE of the LIN-CLUSTER's first LIN-PHYSICAL-CHANNEL, a LinSMSchedule container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the LIN-SCHEDULE-TABLE.</p> <p>LinSMScheduleIndexRef references the LinIfScheduleTable container in the LinIf module configuration, which has been created for the LIN-SCHEDULE-TABLE.</p>

### 3.4.38. LinTp

Configuration parameters	Mapping description
LinTpGlobalConfig	<p>The following parameters are set using the first LIN-TP-NODE which belongs to the imported ECU-INSTANCE. A LIN-TP-NODE belongs to the ECU-INSTANCE if at least one of the COMMUNICATION-CONNECTOR elements it references also belongs to this ECU-INSTANCE.</p> <p>LinTpP2Timing is set to P-2-TIMING.</p> <p>LinTpP2Max is set to P-2-MAX.</p> <p>LinTpMaxNumberOfRespPendingFrames is set to MAX-NUMBER-OF-RESPENDING-FRAMES.</p> <p>If inconsistencies among multiple LIN-TP-NODE elements are detected, a warning is reported.</p>
LinTpGlobalCon- fig/LinTpRxNSdu	<p>For every PDU which is referenced by a LIN-TP-CONNECTION which belongs to the imported ECU-INSTANCE, and which is also received or routed by the ECU-INSTANCE, a LinTpRxNSdu container is created. For more information, see</p>

Configuration parameters	Mapping description
	<p><a href="#">Section 3.3.6, “PDU routing”</a>. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>A LIN-TP-CONNECTION belongs to the imported ECU-INSTANCE if at least one transmitting or receiving LIN-TP-NODE of this LIN-TP-CONNECTION references a COMMUNICATION-CONNECTOR which is also referenced by the imported ECU-INSTANCE.</p> <p>If multiple LIN-TP-CONNECTION elements reference the same PDU, only one LinTpRxNSdu container is created. The parameters TimeoutAs, TimeoutCr, and TimeoutCs of these LIN-TP-CONNECTION elements are checked for consistency. If inconsistencies are detected, a warning is reported.</p> <p>LinTpRxNSduPduRef references the corresponding container in the EcuC module configuration.</p> <p>LinTpRxNSduChannelRef references the container created for the LIN-PHYSICAL-CHANNEL of the LIN-TP-CONNECTION in the LinIf module configuration.</p> <p>LinTpRxNSduTpChannelRef references the LinTpChannelConfig container created for the LIN-PHYSICAL-CHANNEL of the LIN-TP-CONNECTION.</p> <p>LinTpRxNSduNad is set to TP-ADDRESS referenced by the LIN-TP-NODE referenced via TRANSMITTER-REF.</p> <p>LinTpDl is set to LENGTH.</p> <p>LinTpNcr is set to TIMEOUT-CR.</p>
LinTpGlobalConfig/LinTpTxNSdu	<p>For every PDU which is referenced by a LIN-TP-CONNECTION which belongs to the imported ECU-INSTANCE, and which is also sent or routed by the ECU-INSTANCE, a LinTpTxNSdu container is created. For more information, see <a href="#">Section 3.3.6, “PDU routing”</a>. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>If multiple LIN-TP-CONNECTION elements reference the same PDU, only one LinTpTxNSdu container is created. The parameters TimeoutAs, TimeoutCr, and TimeoutCs of these LIN-TP-CONNECTION elements are checked for consistency. If inconsistencies are detected, a warning is reported.</p> <p>LinTpTxNSduPduRef references the corresponding container in the EcuC module configuration.</p>



Configuration parameters	Mapping description
	<p><code>LinTpTxNSduChannelRef</code> references the container created for the LIN-PHYSICAL-CHANNEL of the LIN-TP-CONNECTION in the <code>LinIf</code> module configuration.</p> <p><code>LinTpTxNSduTpChannelRef</code> references the <code>LinTpChannelConfig</code> container created for the LIN-PHYSICAL-CHANNEL of the LIN-TP-CONNECTION.</p> <p><code>LinTpTxNSduNad</code> is set to TP-ADDRESS referenced via MULTICAST-REF. If no MULTICAST-REF exists, the TP-ADDRESS of the first LIN-TP-NODE referenced via RECEIVER-REFS is used. If more than one MULTICAST-REF exists, a warning is issued.</p> <p><code>LinTpNas</code> is set to TIMEOUT-AS.</p> <p><code>LinTpNcs</code> is set to TIMEOUT-CS.</p>
<code>LinTpGlobalConfig/LinTpChannelConfig</code>	<p>For the LIN-PHYSICAL-CHANNEL which belongs to the LIN-TP-CONNECTION a <code>LinTpChannelConfig</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the LIN-PHYSICAL-CHANNEL.</p> <p>A LIN-PHYSICAL-CHANNEL belongs to a LIN-TP-CONNECTION if it is part of the COMMUNICATION-CLUSTER referenced by the parent LIN-TP-CONFIG.</p> <p><code>LinTpDropNotRequestedNad</code> is set to DROP-NOT-REQUESTED-NAD. If inconsistencies are detected among DROP-NOT-REQUESTED-NAD values of multiple LIN-TP-CONNECTION elements which belong to the same LIN-PHYSICAL-CHANNEL, a warning is reported.</p>

### 3.4.39. Nm

Configuration parameters	Mapping description
<code>NmChannelConfig</code>	<p>For every CAN-CLUSTER, J-1939-CLUSTER, FLEXRAY-CLUSTER, and ETHERNET-PHYSICAL-CHANNEL for which an Nm channel container has been created in <code>CanNm</code>, <code>J1939Nm</code>, <code>FrNm</code>, or <code>UdpNm</code>, an <code>NmChannelConfig</code> container is created. For details on Nm channel container creation in the bus-specific NM modules, see <a href="#">Section 3.4.4, “CanNm”</a>, <a href="#">Section 3.4.31, “J1939Nm”</a>, <a href="#">Section 3.4.25, “FrNm”</a>, and <a href="#">Section 3.4.48, “UdpNm”</a>.</p>

Configuration parameters	Mapping description
	<p>The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the NM-CLUSTER or, if the NmChannelConfig has been created for an ETHERNET-PHYSICAL-CHANNEL, the ETHERNET-PHYSICAL-CHANNEL's SHORT-NAME.</p> <p>The NM-ECU used for configuring some of the NmChannelConfig parameters is the NM-ECU which the first NM-NODE that is associated to the configured ECU-INSTANCE and to the NM-CLUSTER references via NM-IF-ECU-REF.</p> <p>NmNodeDetectionEnabled is set to NM-CLUSTER/NM-NODE-DETECTION-ENABLED, or to NM-ECU/NM-NODE-DETECTION-ENABLED if NM-CLUSTER/NM-NODE-DETECTION-ENABLED is not available.</p> <p>NmNodeIdEnabled is set to NM-CLUSTER/NM-NODE-ID-ENABLED, or to NM-ECU/NM-NODE-ID-ENABLED if NM-CLUSTER/NM-NODE-ID-ENABLED is not available.</p> <p>NmRepeatMsgIndEnabled is set to NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED, or to NM-ECU/NM-REPEAT-MSG-IND-ENABLED if NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED is not available.</p> <p>NmSynchronizingNetwork is set to NM-SYNCHRONIZING-NETWORK.</p> <p>NmChannelSleepMaster is set to NM-CHANNEL-SLEEP-MASTER.</p> <p>NmComMChannelRef references the ComMChannel container in the ComM module that corresponds to this NmChannelConfig container.</p> <p>NmBusType/NmStandardBusNmConfig/NmStandardBusType is configured for the following sub types of NM-CLUSTER:</p> <ul style="list-style-type: none"> <li>▶ NM_BUSNM_CANNM for CAN-NM-CLUSTER elements.</li> <li>▶ NM_BUSNM_FRNM for FLEXRAY-NM-CLUSTER elements.</li> <li>▶ NM_BUSNM_UDPNM for UDP-NM-CLUSTER elements.</li> </ul> <p>If the NM-CLUSTER is of type J-1939-NM-CLUSTER, NmBusType/NmGenericBusNmConfig/NmGenericBusNmPrefix is set to J1939Nm.</p> <p>The following parameters are set using the NM-COORDINATOR which belongs to the first NM-NODE belonging to the NM-CLUSTER and the imported ECU-INSTANCE. An NM-COORDINATOR belongs to an NM-NODE if it is aggregated by the NM-ECU of the NM-NODE and if the NM-COORDINATOR references this NM-NODE.</p>

Configuration parameters	Mapping description
	<p>NmShutdownDelayTimer is set to NM-SHUTDOWN-DELAY-TIMER.</p> <p>The following parameters are set using the first NM-NODE which belongs to the NM-CLUSTER and the imported ECU-INSTANCE.</p> <p>NmPassiveModeEnabled is set to NM-PASSIVE-MODE-ENABLED.</p> <p>NmActiveCoordinator is set depending on the value of NM-COORDINATOR-ROLE:</p> <ul style="list-style-type: none"> <li>▶ true for ACTIVE.</li> <li>▶ false for PASSIVE.</li> </ul> <p>NmCoordClusterIndex is set to NM-COORD-CLUSTER. If NM-COORD-CLUSTER is not available, NmCoordClusterIndex is set to INDEX of the NM-COORDINATOR.</p>
NmGlobalConfig	The following parameters are set using the first NM-ECU of all NM-CLUSTER elements belonging to the imported ECU-INSTANCE.
NmGlobalConfig/Nm-GlobalProperties	NmCycletimeMainFunction is set to NM-CYCLETIME-MAIN-FUNCTION.
NmGlobalConfig/Nm-GlobalFeatures	<p>NmUserDataEnabled is set to NM-USER-DATA-ENABLED.</p> <p>NmComUserDataSupport is set to true if any NmUserDataPdu container has been created in the EcuC module configuration as described in <a href="#">Section 3.4.14, "EcuC"</a>, or if any NM-CLUSTER linked to the imported ECU-INSTANCE has its NM-PNC-PARTICIPATION either not defined or set to true.</p> <p>NmPduRxIndicationEnabled is set to NM-PDU-RX-INDICATION-ENABLED.</p> <p>NmStateChangeIndEnabled is set to NM-STATE-CHANGE-IND-ENABLED.</p> <p>NmRemoteSleepIndEnabled is set to NM-REMOTE-SLEEP-IND-ENABLED.</p> <p>NmBusSynchronizationEnabled is set to NM-BUS-SYNCHRONIZATION-ENABLED.</p> <p>NmCoordinatorSupportEnabled is set to true if NM-COORDINATOR elements exist for the NM-ECU elements.</p> <p>NmCarWakeUpRxEnabled is set to true, if at least one NM-CLUSTER belonging to the imported ECU-INSTANCE has NM-CAR-WAKE-UP-RX-ENABLED set to true. If all NM-CLUSTER elements have NM-CAR-WAKE-UP-RX-ENABLED set</p>

Configuration parameters	Mapping description
	<p>to false, NmCarWakeUpRxEnabled is also set to false. Otherwise NmCarWakeUpRxEnabled is not set.</p> <p>The following parameters are set using the first NM-COORDINATOR of all NM-ECU elements of all NM-CLUSTER elements which belong to the imported ECU-INSTANCE.</p> <p>NmGlobalCoordinatorTime is set to NM-GLOBAL-COORDINATOR-TIME.</p> <p>NmCoordinatorSyncSupport is set to NM-COORD-SYNC-SUPPORT.</p>

### 3.4.40. PduR

Configuration parameters	Mapping description
PduRRoutingTables/PduRRoutingTable/PduRRoutingPath	<p>For every PDU instance (see <a href="#">Section 3.3.1.3, "Instance handling"</a>) sent, received or routed (see <a href="#">Section 3.3.6, "PDU routing"</a>) by the imported ECU-INSTANCE, a PduRRoutingPath container is created.</p> <p>For every PDU referenced by a TP-CONNECTION belonging to the imported ECU-INSTANCE, a PduRRoutingPath container is created as well. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU. For NM-PDUs the container name is &lt;PREFIX&gt;&lt;name&gt;_NmComUserData&lt;INSTSUFFIX&gt;.</p> <p>Also, for all Pdu containers created for PNC-enabled NM-CLUSTER elements in the EcuC module (see <a href="#">Section 3.4.14, "EcuC"</a>), a PduRRoutingPath is created. The container name is the name of the Pdu container in the EcuC module, and both PduRSrcPduRef and PduRDestPduRef reference this container.</p> <p>For the following PDU instances no PduRRoutingPath is created:</p> <ul style="list-style-type: none"> <li>▶ PDU instances representing N-PDU elements unless an I-PDU-MAPPING exists for them.</li> <li>▶ PDU instances representing NM-PDU elements for which no NmUserDataPdu container has been created, see <a href="#">Section 3.4.14, "EcuC"</a>.</li> <li>▶ PDU instances sent by the imported ECU-INSTANCE for which also an I-PDU-MAPPING exists which references the PDU's PDU-TRIGGERING via TARGET-I-PDU-REF.</li> </ul>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ PDU instances representing I-SIGNAL-I-PDU elements which do not contain any signal which the ECU-INSTANCE sends or receives and for which no I-PDU-MAPPING exists. If such a PDU is detected, a warning is reported.</li> <li>▶ PDU instances representing DCM-I-PDU elements that are referenced via DO-IP-TP-CONNECTION elements and which are sent and have their DIAG-PDU-TYPE value set to DIAG-REQUEST, or which are received and have their DIAG-PDU-TYPE value set to DIAG-RESPONSE.</li> </ul> <p>PduRSrcPdu/PduRSrcPduRef references the PDU container in the EcuC module configuration. For NM-PDU elements it references the NmUserDataPdu container in the EcuC module which has been created for this NM-PDU. The name of PduRSrcPdu is set to &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;_S, where &lt;name&gt; is the SHORT-NAME of the PDU.</p>
PduRRoutingTables/PduRRoutingTable/PduRRoutingPath/PduRDestPdu	<p>If the PDU instance is routed locally (i.e. the ECU-INSTANCE processes the data contained in the PDU), a PduRDestPdu container is created. Received PDUs which are referenced by TP-CONNECTION elements and by the SOURCE-I-PDU-REF of at least one I-PDU-MAPPING additionally require that one of the following conditions is met:</p> <ul style="list-style-type: none"> <li>▶ The TP-CONNECTION element of the PDU references a multicast TP-ADDRESS</li> <li>▶ The TP-CONNECTION element of the PDU is referenced by a DIAGNOSTIC-CONNECTION which in turn is referenced by a DIAGNOSTIC-SERVICE-TABLE that also references the configured ECU-INSTANCE</li> </ul> <p>The container name is the name of its PduRRoutingPath parent container plus the suffix _D. PduRDestPdu/PduRDestPduRef references the same EcuC PDU container as PduRSrcPdu/PduRSrcPduRef of the parent container.</p> <p>The PDU instance is routed via gateway if two conditions hold:</p> <ul style="list-style-type: none"> <li>▶ The ECU-INSTANCE receives the PDU instance</li> <li>▶ I-PDU-MAPPING elements exist which reference the PDU instance's PDU-TRIGGERING via SOURCE-I-PDU-REF</li> </ul> <p>If the PDU instance is routed via gateway, one PduRDestPdu container is created for each I-PDU-MAPPING. The sent PDU instance that corresponds to the PDU-TRIGGERING that is referenced via I-PDU-MAPPING/TARGET-I-PDU/TARGET-I-PDU-REF is used to configure the PduRDestPdu container. If the I-PDU-MAPPING contains a valid PDUR-TP-CHUNK-SIZE subelement and the PduRRoutingPath does not contain additional sibling PduRDestPdu containers, the content of PDUR-TP-CHUNK-SIZE is used to configure PduRDestPdu/PduRTpThreshold.</p>

Configuration parameters	Mapping description
	<code>PduRDestPduRef</code> references the <code>EcuC</code> container created for the sent PDU instance.
<code>PduRRoutingTables/PduRRoutingTable/PduRRoutingPath/PduRDestPdu/PduDefaultValue/PduDefaultValueElement</code>	<p>For every <code>DEFAULT-VALUE-ELEMENT</code> of the <code>TARGET-I-PDU</code>, a <code>PduRDefaultValueElement</code> container is created. The name of the container is <code>PduRDefaultValueElement_&lt;ELEMENT-POSITION&gt;</code>.</p> <p>If <code>ELEMENT-POSITION</code> <math>\geq</math> <code>PDU's LENGTH</code> a warning is reported and no <code>PduRDefaultValueElement</code> is created.</p> <p>If multiple <code>DEFAULT-VALUE-ELEMENT</code> have the same <code>ELEMENT-POSITION</code>, only one <code>PduRDefaultValueElement</code> container is created and a warning is reported.</p> <p>If <code>ELEMENT-BYTE-VALUE</code> does not lie within the interval <code>[0 .. 255]</code>, a warning is reported and no <code>PduRDefaultValueElement</code> is created.</p> <p><code>PduRDefaultValueElement</code> is set to <code>ELEMENT-BYTE-VALUE</code>.</p> <p><code>PduRDefaultValueElementBytePosition</code> is set to <code>ELEMENT-POSITION</code>.</p>
<code>PduRRoutingTables/PduRRoutingPathGroup</code>	<p>For every <code>PDUR-I-PDU-GROUP</code> referenced by the imported <code>ECU-INSTANCE</code> via <code>ASSOCIATED-PDUR-I-PDU-GROUP-REF</code>, a <code>PduRRoutingPathGroup</code> container is created. The name of the container is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>PDUR-I-PDU-GROUP</code>.</p> <p>For each <code>PDU-TRIGGERING-REF</code> of the <code>PDUR-I-PDU-GROUP</code> a <code>PduRDestPduRef</code> reference is created. It refers to the <code>PduRDestPdu</code> container that has been created for the <code>PDU-TRIGGERING</code> referenced by <code>PDU-TRIGGERING-REF</code>.</p>

### 3.4.41. Sd

In its R19/11 release, AUTOSAR introduced a new communication and service model for Ethernet that supersedes the model that was present in AUTOSAR 4.4 an earlier releases. The AUTOSAR R19-11 schema allows to provide Ethernet communication and service data using either model. Both models are supported for the configuration of the `Sd` module. It is, however, strongly discouraged to mix AUTOSAR 4.4 and R19-11 communication and service model concepts in a single Ecu Extract or System Description.

This chapter describes the configuration for both AUTOSAR 4.4 and R19-11 model, [5] and [6] provide in their "Ethernet specific description" chapters details on the differences of both models.

In AUTOSAR 4.4, a `PROVIDED-SERVICE-INSTANCE/CONSUMED-SERVICE-INSTANCE` belong to a `SOCKET-ADDRESS` if the `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` that in turn contains the `PROVIDED-SERVICE-INSTANCE/CONSUMED-SERVICE-INSTANCE`.

In AUTOSAR R19-11, a `PROVIDED-SERVICE-INSTANCE` belongs to a `SOCKET-ADDRESS` if it references the `APPLICATION-ENDPOINT` of the `SOCKET-ADDRESS` via `PROVIDED-SERVICE-INSTANCE/LOCAL-ADDRESS`. `CONSUMED-SERVICE-INSTANCE` belongs to a `SOCKET-ADDRESS` if it references the `APPLICATION-ENDPOINT` of the `SOCKET-ADDRESS` via `CONSUMED-SERVICE-INSTANCE/LOCAL-ADDRESS` or if it contains one or more `CONSUMED-EVENT-GROUP` elements which reference the `APPLICATION-ENDPOINT` of the `SOCKET-ADDRESS` via `CONSUMED-EVENT-GROUP/EVENT-MULTICAST-ADDRESS`.

Configuration parameters	Mapping description
SdConfig/SdInstance	<p>For each <code>ETHERNET-PHYSICAL-CHANNEL</code> for which the following conditions hold, one <code>SdInstance</code> container is created.</p> <ul style="list-style-type: none"> <li>▶ The imported <code>ECU-INSTANCE</code> sends and/or receives PDUs on the <code>ETHERNET-PHYSICAL-CHANNEL</code>.</li> <li>▶ The imported <code>ECU-INSTANCE</code> owns on the <code>ETHERNET-PHYSICAL-CHANNEL</code> at least one <i>local</i> <code>SOCKET-ADDRESS</code> to which in turn belongs at least one <code>PROVIDED-SERVICE-INSTANCE</code> or at least one <code>CONSUMED-SERVICE-INSTANCE</code>. For more information about <code>SOCKET-ADDRESS</code>, see <a href="#">Section 3.4.43, “SoAd”</a>.</li> </ul> <p>The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the <code>ETHERNET-PHYSICAL-CHANNEL</code>.</p> <p><code>SdInstanceHostname</code> is configured by retrieving the <code>FULLY-QUALIFIED-DOMAIN-NAME</code> elements of all <code>NETWORK-ENDPOINT</code> elements that belong to the <code>ETHERNET-PHYSICAL-CHANNEL</code> and to an <code>ETHERNET-COMMUNICATION-CONNECTOR</code> of the imported <code>ECU-INSTANCE</code>.</p> <p>A <code>NETWORK-ENDPOINT</code> belongs to the <code>ETHERNET-PHYSICAL-CHANNEL</code> and to an <code>ETHERNET-COMMUNICATION-CONNECTOR</code> if there is a <code>SOCKET-ADDRESS</code> contained in the <code>ETHERNET-PHYSICAL-CHANNEL</code> which references the <code>NETWORK-ENDPOINT</code> via <code>SOCKET-ADDRESS/NETWORK-ENDPOINT-REF</code> and which contains an <code>APPLICATION-ENDPOINT</code> that in turn references the <code>ETHERNET-COMMUNICATION-CONNECTOR</code> of the <code>ECU-INSTANCE</code> either via <code>APPLICATION-ENDPOINT/CONNECTOR-REF</code> or via <code>APPLICATION-ENDPOINT/MULTICAST-CONNECTOR-REFS/MULTICAST-CONNECTOR-REF</code>.</p> <p>If the collection yields exactly one distinct <code>FULLY-QUALIFIED-DOMAIN-NAME</code>, it is used to configure <code>SdInstanceHostname</code>. If none or more than one distinct <code>FULLY-QUALIFIED-DOMAIN-NAME</code> elements are found, <code>SdInstanceHostname</code> is not set.</p>

Configuration parameters	Mapping description
SdConfig/SdInstance/SdInstanceTxPdu	If an Sd PDU exists which is sent by the imported ECU-INSTANCE via a <i>local</i> SOCKET-ADDRESS of the ETHERNET-PHYSICAL-CHANNEL, an SdInstanceTxPdu container is created. SdTxPduRef references the corresponding container in the EcuC module configuration. A PDU is considered an Sd PDU if it is a GENERAL-PURPOSE-PDU and its CATEGORY is set to SD.
SdConfig/SdInstance/SdInstanceUnicastRxPdu	If an Sd PDU exists which is received by the imported ECU-INSTANCE via a <i>local unicast local</i> SOCKET-ADDRESS of the ETHERNET-PHYSICAL-CHANNEL, an SdInstanceUnicastRxPdu container is created. SdRxPduRef references the corresponding container in the EcuC module configuration.
SdConfig/SdInstance/SdInstanceMulticastRxPdu	If an Sd PDU exists which is received by the imported ECU-INSTANCE a <i>local multicast local</i> SOCKET-ADDRESS of the ETHERNET-PHYSICAL-CHANNEL, an SdInstanceMulticastRxPdu container is created. SdRxPduRef references the corresponding container in the EcuC module configuration.
SdConfig/SdInstance/SdServerService	<p>For every <i>local</i> SOCKET-ADDRESS of the imported ECU-INSTANCE all its PROVIDED-SERVICE-INSTANCE elements are retrieved. For each of these PROVIDED-SERVICE-INSTANCE elements, an SdServerService container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the PROVIDED-SERVICE-INSTANCE.</p> <p>SdServerServiceInstanceId is set to INSTANCE-IDENTIFIER.</p> <p>SdServerServiceId is set to SERVICE-IDENTIFIER.</p> <p>SdServerServiceMajorVersion is set to SD-SERVER-CONFIG/SERVER-SERVICE-MAJOR-VERSION in AUTOSAR 4.4 models and to MAJOR-VERSION in AUTOSAR R19-11 models.</p> <p>SdServerServiceMinorVersion is set to SD-SERVER-CONFIG/SERVER-SERVICE-MINOR-VERSION in AUTOSAR 4.4 models and to MINOR-VERSION in AUTOSAR R19-11 models.</p> <p>SdServerServiceAutoAvailable is set to PROVIDED-SERVICE-INSTANCE/AUTO-AVAILABLE in AUTOSAR R19-11 models.</p> <p>SdServerServiceTimerRef references the SdServerTimer container created either for the AUTOSAR 4.4 model SD-SERVER-CONFIG, or for the AUTOSAR R19-11 SOMEIP-SD-SERVER-SERVICE-INSTANCE-CONFIG.</p> <p>If the <i>local</i> SOCKET-ADDRESS is connected to a SoAdSocketConnection-Group of type SoAdSocketUdp and the SOCKET-ADDRESS is a unicast address, SdServerServiceUdpRef references the SoAdSocketConnection-</p>



Configuration parameters	Mapping description
	<p>Group. For more information about SOCKET-ADDRESS, see <a href="#">Section 3.4.43, “SoAd”</a>.</p> <p>If the <i>local</i> SOCKET-ADDRESS is connected to a SoAdSocketConnection-Group of type SoAdSocketTcp and the SOCKET-ADDRESS is a unicast address, SdServerServiceTcpRef references the SoAdSocketConnection-Group.</p>
SdConfig/SdInstance/SdServerService/SdServerCapabilityRecord	<p>For each CAPABILITY-RECORDS/TAG-WITH-OPTIONAL-VALUE, an SdServerCapabilityRecord container is created. The container name is SdServerCapabilityRecord_&lt;auto incremented number&gt;.</p> <p>SdServerCapabilityRecordKey is set to KEY.</p> <p>SdServerCapabilityRecordValue is set to VALUE.</p>
SdConfig/SdInstance/SdServerService/SdEventHandler	<p>For each EVENT-HANDLER of the PROVIDED-SERVICE-INSTANCE, an SdEventHandler container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the EVENT-HANDLER.</p> <p>SdEventHandlerEventGroupId is set to EVENT-GROUP-IDENTIFIER of the first CONSUMED-EVENT-GROUP that is referenced by the EVENT-HANDLER in AUTOSAR 4.4 models, and to the EVENT-GROUP-IDENTIFIER of the EVENT-HANDLER in AUTOSAR R19-11 models.</p> <p>SdEventHandlerMulticastThreshold is set to EVENT-HANDLER/MULTICAST-THRESHOLD.</p> <p>If any of the AUTOSAR 4.4 SO-AD-ROUTING-GROUP elements referenced by the EVENT-HANDLER or any of the AUTOSAR 19-11 PDU-ACTIVATION-ROUTING-GROUP elements aggregated in the EVENT-HANDLER has its EVENT-GROUP-CONTROL-TYPE set to ACTIVATION-MULTICAST, SdEventHandlerMulticast/SdEventActivationRef references the SoAdRoutingGroup container that has been created for this SO-AD-ROUTING-GROUP or PDU-ACTIVATION-ROUTING-GROUP.</p> <p>In AUTOSAR 4.4, the APPLICATION-ENDPOINT that transmits the multicast PDUs of an EVENT-HANDLER is either the APPLICATION-ENDPOINT that EVENT-HANDLER/APPLICATION-ENDPOINT-REF refers to, or, if EVENT-HANDLER/APPLICATION-ENDPOINT-REF does not exist, the APPLICATION-ENDPOINT that contains the EVENT-HANDLER. In AUTOSAR 19-11 the APPLICATION-ENDPOINT is the one that the parent PROVIDED-SERVICE-INSTANCE of the EVENT-HANDLER references via PROVIDED-SERVICE-INSTANCE/LOCAL-UNICAST-ADDRESS and which represents a Udp address. SdMulti-</p>

Configuration parameters	Mapping description
	<p><code>castEventSoConRef</code> references a <i>multicast</i> <code>SoAdSocketConnection</code> container in the <code>SoAd</code> module if the <code>APPLICATION-ENDPOINT</code> meets the following conditions:</p> <ul style="list-style-type: none"> <li>▶ The <code>APPLICATION-ENDPOINT</code> is aggregated by a <i>local</i> <code>SOCKET-ADDRESS</code> of the imported <code>ECU-INSTANCE</code>.</li> <li>▶ The <code>SoAd</code> configuration contains a <code>SoAdSocketConnectionGroup</code> container that uses the <code>SOCKET-ADDRESS</code> as <i>local</i> address.</li> <li>▶ One of the following conditions is met: <ul style="list-style-type: none"> <li>▶ The <code>SoAdSocketConnectionGroup</code> configuration contains a single <code>SoAdSocketConnection</code> that is associated with a <i>remote multicast</i> <code>SOCKET-ADDRESS</code>.</li> <li>▶ In AUTOSAR 4.4 models, the <code>SoAdSocketConnectionGroup</code> configuration contains two or more <code>SoAdSocketConnection</code> containers that are associated with a <i>remote multicast</i> <code>SOCKET-ADDRESS</code>, but only one of the <code>SoAdSocketConnection</code> containers refers to the <i>remote multicast</i> <code>SOCKET-ADDRESS</code> that all <code>CONSUMED-EVENT-GROUP</code> elements of the <code>EVENT-HANDLER</code> refer to as well.</li> </ul> </li> </ul> <p>The <i>multicast</i> <code>SOCKET-ADDRESS</code> of a <code>CONSUMED-EVENT-GROUP</code> is the <code>SOCKET-ADDRESS</code> that contains the <code>APPLICATION-ENDPOINT</code> that the <code>CONSUMED-EVENT-GROUP</code> refers to via <code>APPLICATION-ENDPOINT-REF</code>.</p> <p><code>SdEventHandlerTimerRef</code> is configured in the same way as <code>SdServerServiceTimerRef</code> in AUTOSAR 4.4 models. In AUTOSAR R19-11 models, <code>SdEventHandlerTimerRef</code> references the <code>SdServerTimer</code> container created for the AUTOSAR R19-11 <code>SOMEIP-SD-SERVER-EVENT-GROUP-TIMING-CONFIG</code>.</p> <p><code>SdServerCapabilityRecord</code> is configured in the same way as <code>SdServerService/SdServerCapabilityRecord</code> in AUTOSAR 4.4 models. In AUTOSAR R19-11 there are no capability records for <code>EVENT-HANDLER</code> elements, hence nothing is configured.</p>
<code>SdConfig/SdInstance/SdServerService/SdEventHandler/SdEventHandlerUdp</code>	<p>If the <i>local</i> <code>SOCKET-ADDRESS</code> is connected to a <code>SoAdSocketConnectionGroup</code> of type <code>SoAdSocketUdp</code> (see <a href="#">Section 3.4.43, “SoAd”</a>), an <code>SdEventHandlerUdp</code> container is created.</p> <p>If any of the AUTOSAR 4.4 <code>SO-AD-ROUTING-GROUP</code> elements or any of the AUTOSAR R19-11 <code>PDU-ACTIVATION-ROUTING-GROUP</code> elements of</p>

Configuration parameters	Mapping description
	<p>the EVENT-HANDLER has its EVENT-GROUP-CONTROL-TYPE set to ACTIVATION-UNICAST or to ACTIVATION-AND-TRIGGER-UNICAST, SdEventActivationRef references the SoAdRoutingGroup container that has been created for this SO-AD-ROUTING-GROUP or PDU-ACTIVATION-ROUTING-GROUP.</p> <p>If any of the AUTOSAR 4.4 SO-AD-ROUTING-GROUP or any of the AUTOSAR R19-11 PDU-ACTIVATION-ROUTING-GROUP elements of the EVENT-HANDLER has its EVENT-GROUP-CONTROL-TYPE set to TRIGGER-UNICAST or to ACTIVATION-AND-TRIGGER-UNICAST, SdEventTriggeringRef references the SoAdRoutingGroup container that has been created for this SO-AD-ROUTING-GROUP.</p>
SdConfig/SdInstance/SdServerService/SdEventHandler/SdEventHandlerTcp	<p>If the <i>local</i> SOCKET-ADDRESS is connected to a SoAdSocketConnectionGroup of type SoAdSocketTcp (see <a href="#">Section 3.4.43, "SoAd"</a>), an SdEventHandlerTcp container is created.</p> <p>The parameters SdEventActivationRef and SdEventTriggeringRef are configured in the same way as the corresponding parameters of SdEventHandlerUdp.</p>
SdConfig/SdInstance/SdServerService/SdProvidedMethods	<p>SdServerServiceActivationRef references the SoAdRoutingGroup container that has been created for the first AUTOSAR 4.4 SO-AD-ROUTING-GROUP referenced by the PROVIDED-SERVICE-INSTANCE or the AUTOSAR R19-11 PDU-ACTIVATION-ROUTING-GROUP that the PROVIDED-SERVICE-INSTANCE contains in METHOD-ACTIVATION-ROUTING-GROUPS.</p>
SdConfig/SdInstance/SdServerTimer	<p>SdServerTimer are created for different AUTOSAR 4.4 and AUTOSAR R19-11 elements. The container name is SdServerTimer&lt;auto incremented number&gt;.</p> <p>► For all AUTOSAR 4.4 SD-SERVER-CONFIG elements, the following parameters are configured:</p> <p>SdServerTimerInitialOfferDelayMax is set to INITIAL-OFFER-BEHAVIOR/INITIAL-DELAY-MAX-VALUE.</p> <p>SdServerTimerInitialOfferDelayMin is set to INITIAL-OFFER-BEHAVIOR/INITIAL-DELAY-MIN-VALUE.</p> <p>SdServerTimerInitialOfferRepetitionBaseDelay is set to INITIAL-OFFER-BEHAVIOR/INITIAL-REPETITIONS-BASE-DELAY.</p>

Configuration parameters	Mapping description
	<p>SdServerTimerInitialOfferRepetitionsMax <b>is set to</b> INITIAL-OFFER-BEHAVIOR/INITIAL-REPETITIONS-MAX.</p> <p>SdServerTimerOfferCyclicDelay <b>is set to</b> OFFER-CYCLIC-DELAY.</p> <p>SdServerTimerRequestResponseMaxDelay <b>is set to</b> REQUEST-RESPONSE-DELAY/MAX-VALUE.</p> <p>SdServerTimerRequestResponseMinDelay <b>is set to</b> REQUEST-RESPONSE-DELAY/MIN-VALUE.</p> <p>SdServerTimerTTL <b>is set to</b> TTL.</p> <p>► For all AUTOSAR R19-11 SOMEIP-SD-SERVER-SERVICE-INSTANCE-CONFIG elements, the following parameters are configured:</p> <p>SdServerTimerInitialOfferDelayMax <b>is set to</b> INITIAL-OFFER-BEHAVIOR/INITIAL-DELAY-MAX-VALUE.</p> <p>SdServerTimerInitialOfferDelayMin <b>is set to</b> INITIAL-OFFER-BEHAVIOR/INITIAL-DELAY-MIN-VALUE.</p> <p>SdServerTimerInitialOfferRepetitionBaseDelay <b>is set to</b> INITIAL-OFFER-BEHAVIOR/INITIAL-REPETITIONS-BASE-DELAY.</p> <p>SdServerTimerInitialOfferRepetitionsMax <b>is set to</b> INITIAL-OFFER-BEHAVIOR/INITIAL-REPETITIONS-MAX.</p> <p>SdServerTimerOfferCyclicDelay <b>is set to</b> OFFER-CYCLIC-DELAY.</p> <p>SdServerTimerRequestResponseMaxDelay <b>is set to</b> REQUEST-RESPONSE-DELAY/MAX-VALUE.</p> <p>SdServerTimerRequestResponseMinDelay <b>is set to</b> REQUEST-RESPONSE-DELAY/MIN-VALUE.</p> <p>SdServerTimerTTL <b>is set to</b> SERVICE-OFFER-TIME-TO-LIVE.</p> <p>► For all AUTOSAR R19-11 SOMEIP-SD-SERVER-EVENT-GROUP-TIMING-CONFIG elements, the following parameters are configured:</p> <p>SdServerTimerRequestResponseMaxDelay <b>is set to</b> REQUEST-RESPONSE-DELAY/MAX-VALUE.</p>

Configuration parameters	Mapping description
	<p><code>SdServerTimerRequestResponseMinDelay</code> is set to REQUEST-RESPONSE-DELAY/MIN-VALUE.</p>
<p><code>SdConfig/SdInstance/SdClientService</code></p>	<p>For every local SOCKET-ADDRESS of the imported ECU-INSTANCE all its CONSUMED-SERVICE-INSTANCE elements are retrieved. For each of these CONSUMED-SERVICE-INSTANCE elements, an <code>SdClientService</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code> where <code>&lt;name&gt;</code> is the SHORT-NAME of the CONSUMED-SERVICE-INSTANCE.</p> <p>The parameters <code>SdClientServiceInstanceId</code>, <code>SdClientServiceId</code>, <code>SdClientServiceMajorVersion</code>, <code>SdClientServiceMinorVersion</code>, <code>SdClientServiceTimerRef</code> and <code>SdClientCapabilityRecord</code> are configured in the same way as the corresponding parameters of the <code>SdServerService</code> container.</p> <p><code>SdClientServiceAutoRequire</code> is set to CONSUMED-SERVICE-INSTANCE/AUTO-REQUIRE in AUTOSAR R19-11 models.</p> <p>If the <i>local</i> SOCKET-ADDRESS is connected to a <code>SoAdSocketConnectionGroup</code> of type <code>SoAdSocketUdp</code> and the SOCKET-ADDRESS is a unicast address, <code>SdClientServiceUdpRef</code> references this <code>SoAdSocketConnectionGroup</code>. For more information about SOCKET-ADDRESS, see <a href="#">Section 3.4.43, “SoAd”</a>.</p> <p>If the <i>local</i> SOCKET-ADDRESS is connected to a <code>SoAdSocketConnectionGroup</code> of type <code>SoAdSocketTcp</code> and the SOCKET-ADDRESS is a unicast address, <code>SdClientServiceTcpRef</code> references this <code>SoAdSocketConnectionGroup</code>.</p>
<p><code>SdConfig/SdInstance/SdClientService/SdConsumedEventGroup</code></p>	<p>For each CONSUMED-EVENT-GROUP of the CONSUMED-SERVICE-INSTANCE, an <code>SdConsumedEventGroup</code> container is created. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the CONSUMED-EVENT-GROUP.</p> <p><code>SdConsumedEventGroupId</code> is set to EVENT-GROUP-IDENTIFIER.</p> <p><code>SdConsumedEventGroupTimerRef</code> references the <code>SdClientTimer</code> container created either for the AUTOSAR 4.4 model <code>SD-CLIENT-CONFIG</code>, or for the AUTOSAR R19-11 <code>SOMEIP-SD-CLIENT-SERVICE-INSTANCE-CONFIG</code>.</p> <p><code>SdClientCapabilityRecord</code>: see <code>SdServerService/SdServerCapabilityRecord</code>.</p>

Configuration parameters	Mapping description
	<p>If any of the AUTOSAR 4.4 SO-AD-ROUTING-GROUP elements referenced by the CONSUMED-EVENT-GROUP or any of the AUTOSAR R19-11 PDU-ACTIVATION-ROUTING-GROUP elements aggregated in the CONSUMED-EVENT-GROUP has its EVENT-GROUP-CONTROL-TYPE set to ACTIVATION-UNICAST or to ACTIVATION-AND-TRIGGER-UNICAST, a reference is set up to the SoAdRoutingGroup container that has been created for this SO-AD-ROUTING-GROUP or PDU-ACTIVATION-ROUTING-GROUP. If the <i>local</i> SOCKET-ADDRESS is connected to a SoAdSocketConnectionGroup of type SoAdSocketUdp, the reference is SdConsumedEventGroupUdpActivationRef, if the SoAdSocketConnectionGroup is of type SoAdSocketTcp the reference is SdConsumedEventGroupTcpActivationRef. For more information about SOCKET-ADDRESS, see <a href="#">Section 3.4.43, “SoAd”</a>.</p> <p>If any of the AUTOSAR 4.4 SO-AD-ROUTING-GROUP elements or any of the AUTOSAR R19-11 PDU-ACTIVATION-ROUTING-GROUP elements of the CONSUMED-EVENT-GROUP has its EVENT-GROUP-CONTROL-TYPE set to ACTIVATION-MULTICAST, SdConsumedEventGroupMulticastActivationRef references the SoAdRoutingGroup container that has been created for this SO-AD-ROUTING-GROUP.</p> <p>SdConsumedEventGroupMulticastGroupRef is set up to reference a SoAdSocketConnectionGroup container in the SoAd module if the following conditions are met:</p> <ul style="list-style-type: none"> <li>▶ The CONSUMED-EVENT-GROUP references an APPLICATION-ENDPOINT via APPLICATION-ENDPOINT-REF in AUTOSAR 4.4, or via EVENT-MULTICAST-ADDRESS in AUTOSAR R19-11. The APPLICATION-ENDPOINT is aggregated by a <i>local multicast</i> SOCKET-ADDRESS of the imported ECU-INSTANCE.</li> <li>▶ The SoAd configuration contains a SoAdSocketConnectionGroup container that uses the SOCKET-ADDRESS as <i>local</i> address.</li> </ul>
SdConfig/SdInstance/SdClientService/SdConsumed-Methods	SdClientServiceActivationRef references the SoAdRoutingGroup container that has been created for the first AUTOSAR 4.4 SO-AD-ROUTING-GROUP referenced by the CONSUMED-SERVICE-INSTANCE or for the first AUTOSAR R19-11 PDU-ACTIVATION-ROUTING-GROUP contained in the CONSUMED-SERVICE-INSTANCE.
SdConfig/SdInstance/SdClient-Timer	SdClientTimer are created for different AUTOSAR 4.4 and AUTOSAR R19-11 elements. The container name is SdClientTimer<auto incremented number>.

Configuration parameters	Mapping description
	<p>► For all AUTOSAR 4.4 SD-CLIENT-CONFIG elements, the following parameters are configured:</p> <p>SdClientTimerInitialFindDelayMax is set to INITIAL-FIND-BEHAVIOR/INITIAL-DELAY-MAX-VALUE.</p> <p>SdClientTimerInitialFindDelayMin is set to INITIAL-FIND-BEHAVIOR/INITIAL-DELAY-MIN-VALUE.</p> <p>SdClientTimerInitialFindRepetitionsBaseDelay is set to INITIAL-FIND-BEHAVIOR/INITIAL-REPETITIONS-BASE-DELAY.</p> <p>SdClientTimerInitialFindRepetitionsMax is set to INITIAL-FIND-BEHAVIOR/INITIAL-REPETITIONS-MAX.</p> <p>SdClientTimerRequestResponseMaxDelay is set to REQUEST-RESPONSE-DELAY/MAX-VALUE.</p> <p>SdClientTimerRequestResponseMinDelay is set to REQUEST-RESPONSE-DELAY/MIN-VALUE.</p> <p>SdClientTimerTTL is set to TTL.</p> <p>► For all AUTOSAR R19-11 SOMEIP-SD-CLIENT-EVENT-GROUP-TIMING-CONFIG elements, the following parameters are configured:</p> <p>SdClientTimerRequestResponseMaxDelay is set to REQUEST-RESPONSE-DELAY/MAX-VALUE.</p> <p>SdClientTimerRequestResponseMinDelay is set to REQUEST-RESPONSE-DELAY/MIN-VALUE.</p> <p>SdClientTimerTTL is set to TIME-TO-LIVE.</p> <p>► For all AUTOSAR R19-11 SOMEIP-SD-CLIENT-SERVICE-INSTANCE-CONFIG elements, the following parameters are configured:</p> <p>SdClientTimerInitialFindDelayMax is set to INITIAL-FIND-BEHAVIOR/INITIAL-DELAY-MAX-VALUE.</p> <p>SdClientTimerInitialFindDelayMin is set to INITIAL-FIND-BEHAVIOR/INITIAL-DELAY-MIN-VALUE.</p> <p>SdClientTimerInitialFindRepetitionsBaseDelay is set to INITIAL-FIND-BEHAVIOR/INITIAL-REPETITIONS-BASE-DELAY.</p>

Configuration parameters	Mapping description
	<p>SdClientTimerInitialFindRepetitionsMax is set to INITIAL-FIND-BEHAVIOR/INITIAL-REPETITIONS-MAX.</p> <p>SdClientTimerTTL is set to SERVICE-FIND-TIME-TO-LIVE.</p>

### 3.4.42. SecOC

The SecOC secures or authenticates the payload of a PDU by using a SECURED-I-PDU and an associated *payload PDU*. A SECURED-I-PDU is associated with a payload PDU if SECURED-I-PDU/PAYLOAD-REF refers to a PDU-TRIGGERING which in turn refers to the payload PDU.

If the SECURED-I-PDU has its USE-AS-CRYPTOGRAPHIC-I-PDU parameter either set to false or not set at all, the SECURED-I-PDU contains the associated payload PDU and the meta-data, i.e. *Message Authentication Code* and *Freshness Value*.

If the SECURED-I-PDU has its USE-AS-CRYPTOGRAPHIC-I-PDU parameter set to true, the SECURED-I-PDU mostly contains the security data, while the payload is independently transmitted in a second payload PDU. A SECURED-I-PDU that has its USE-AS-CRYPTOGRAPHIC-I-PDU set to true is referred to as *cryptographic PDU*. The PDU that contains the encrypted or authenticated payload is referred to as *payload PDU*.

Configuration parameters	Mapping description
SecOC/SecOCTx-PduProcessing, SecOC/SecOCRxPduProcessing	<p>For every SECURED-I-PDU that the imported ECU-INSTANCE sends, a SecOC-TxPduProcessing container is created, for every SECURED-I-PDU that is received by the imported ECU-INSTANCE, a SecOCRxPduProcessing container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the SECURED-I-PDU.</p> <p>SecOCPduType is set to SECOC_TPPDU in the following cases:</p> <ul style="list-style-type: none"> <li>▶ The SECURED-I-PDU is referenced by a CAN-TP-CONNECTION via TP-SDU-REF</li> <li>▶ The SECURED-I-PDU is referenced by a FLEXRAY-TP-CONNECTION via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF</li> <li>▶ The SECURED-I-PDU is referenced by a FLEXRAY-AR-TP-CONNECTION via DIRECT-TP-SDU-REF or via REVERSED-TP-SDU-REF</li> <li>▶ The SECURED-I-PDU is referenced by a LIN-TP-CONNECTION via LIN-TP-N-SDU-REF</li> </ul>



Configuration parameters	Mapping description
	<p>► The SECURED-I-PDU is referenced by a PDU-TRIGGERING, which in turn is referenced by a SOMEIP-TP-CONNECTION via TP-SDU-REF</p> <p>► An ETH-TP-CONNECTION, a DO-IP-TP-CONNECTION, or a SOMEIP-TP-CONNECTION refers to the PDU-TRIGGERING that references the transmitted SECURED-I-PDU and that belongs to the PHYSICAL-CHANNEL on which the I-SIGNAL-I-PDU is transmitted.</p> <p>In all other cases, the value of SecOCPduType is set to SECOC_IFPDU.</p> <p>If AUTHENTICATION-PROPS-REF refers to a valid SECURE-COMMUNICATION-AUTHENTICATION-PROPS element, SecOCAuthInfoTxLength is set to SECURE-COMMUNICATION-AUTHENTICATION-PROPS/AUTH-INFO-TX-LENGTH. Otherwise SecOCAuthInfoTxLength is set to SECURE-COMMUNICATION-PROPS/AUTH-INFO-TX-LENGTH.</p> <p>SecOCAuthenticationBuildAttempts is set to SECURE-COMMUNICATION-PROPS/AUTHENTICATION-BUILD-ATTEMPTS.</p> <p>SecOCDataId is set to SECURE-COMMUNICATION-PROPS/DATA-ID.</p> <p>SecOCFreshnessValueId is set to SECURE-COMMUNICATION-PROPS/FRESHNESS-VALUE-ID.</p> <p>If FRESHNESS-PROPS-REF refers to a valid SECURE-COMMUNICATION-FRESHNESS-PROPS element, SecOCFreshnessValueLength is set to SECURE-COMMUNICATION-FRESHNESS-PROPS/FRESHNESS-VALUE-LENGTH. Otherwise SecOCFreshnessValueLength is set to SECURE-COMMUNICATION-PROPS/FRESHNESS-VALUE-LENGTH.</p> <p>If FRESHNESS-PROPS-REF refers to a valid SECURE-COMMUNICATION-FRESHNESS-PROPS element, SecOCFreshnessValueTxLength is set to SECURE-COMMUNICATION-FRESHNESS-PROPS/FRESHNESS-VALUE-TX-LENGTH. Otherwise SecOCFreshnessValueTxLength is set to SECURE-COMMUNICATION-PROPS/FRESHNESS-VALUE-TX-LENGTH.</p> <p>SecOCUseAuthDataFreshness is set to USE-AUTH-DATA-FRESHNESS of the I-PDU-PORT via which the SECURED-I-PDU is sent or received.</p>
SecOC/SecOCTx-PduProcessing	<p>For every SECURED-I-PDU that the imported ECU-INSTANCE sends, the following parameters are configured in SecOCTxPduProcessing:</p> <p>► SecOCTxAuthenticPduLayer/SecOCTxAuthenticLayerPduRef is set to reference the EcuC configuration container of the PDU that the SE-</p>

Configuration parameters	Mapping description
	<p>CURED-I-PDU references via PAYLOAD-REF. That container represents the PDU that is provided by the upper layer module. The following PDU types are supported:</p> <ul style="list-style-type: none"> <li>▶ I-SIGNAL-I-PDU</li> <li>▶ CONTAINER-I-PDU</li> <li>▶ MULTIPLEXED-I-PDU</li> <li>▶ USER-DEFINED-I-PDU</li> <li>▶ DCM-I-PDU</li> <li>▶ GENERAL-PURPOSE-I-PDU</li> </ul> <p>▶ SecOCTxPduSecuredArea/SecOCSecuredTxPduLength is set to SECURE-COMMUNICATION-PROPS/SECURED-AREA-LENGTH.</p> <p>▶ SecOCTxPduSecuredArea/SecOCSecuredTxPduOffset is set to SECURE-COMMUNICATION-PROPS/SECURED-AREA-OFFSET.</p> <p>▶ SecOCTxAuthServiceConfigRef is set to reference the CsmJob container that has been created for the SEC-OC-CRYPTO-SERVICE-MAPPING which the PDU-TRIGGERING of the SECURED-I-PDU is referencing.</p> <p>The content of the choice container SecOCTxSecuredPduLayer depends on whether the SECURED-I-PDU is a <i>cryptographic</i> PDU or not. If it is a <i>cryptographic</i> PDU, the content of the choice container is set to SecOCTxSecuredPduCollection, otherwise to SecOCTxSecuredPdu.</p> <p>SecOCTxSecuredPduCollection/SecOCTxAuthenticPdu/SecOCTxAuthenticPduRef is set to reference the configuration container of the <i>payload</i> PDU in the EcuC module configuration.</p> <p>SecOCTxSecuredPduCollection/SecOCTxCryptographicPdu/SecOCTxCryptographicPduRef is set to reference the configuration container of the <i>cryptographic</i> PDU in the EcuC module configuration.</p> <p>SecOCTxSecuredPduCollection/SecOCUseMessageLink/SecOCMessageLinkLen is set to SECURE-COMMUNICATION-PROPS/MESSAGE-LINK-LENGTH, SecOCTxSecuredPduCollection/SecOCUseMessageLink/SecOCMessageLinkPos is set to SECURE-COMMUNICATION-PROPS/MESSAGE-LINK-POSITION.</p> <p>SecOCTxSecuredPdu/SecOCTxSecuredLayerPduRef is set to reference the configuration container of the SECURED-I-PDU in the EcuC module configuration.</p>

Configuration parameters	Mapping description
	<p>SecOCTxSecuredPdu/SecOCAuthPduHeaderLength and SecOCTxSecuredPduCollection/SecOCTxAuthenticPdu/SecOCAuthPduHeaderLength are configured depending on the value of SECURED-I-PDU/USE-SECURED-PDU-HEADER:</p> <ul style="list-style-type: none"> <li>▶ 0 for NO-HEADER</li> <li>▶ 1 for SECURED-PDU-HEADER-08-BIT</li> <li>▶ 2 for SECURED-PDU-HEADER-16-BIT</li> <li>▶ 4 for SECURED-PDU-HEADER-32-BIT</li> </ul>
SecOC/SecOCRx-PduProcessing	<p>For every SECURED-I-PDU that the imported ECU-INSTANCE receives, the following parameters are configured in SecOCRxPduProcessing:</p> <ul style="list-style-type: none"> <li>▶ SecOCAuthenticationVerifyAttempts is set to SECURE-COMMUNICATION-PROPS/AUTHENTICATION-RETRIES.</li> <li>▶ SecOCAuthDataFreshnessLen is set to SECURE-COMMUNICATION-PROPS/AUTH-DATA-FRESHNESS-LENGTH.</li> <li>▶ SecOCAuthDataFreshnessStartPosition is set to SECURE-COMMUNICATION-PROPS/AUTH-DATA-FRESHNESS-START-POSITION.</li> <li>▶ SecOCRxAuthenticPduLayer/SecOCRxAuthenticLayerPduRef is set to reference the EcuC configuration container of the PDU that the SECURED-I-PDU references via PAYLOAD-REF. That container represents the PDU that is provided to the upper layer module. The following PDU types are supported: <ul style="list-style-type: none"> <li>▶ I-SIGNAL-I-PDU</li> <li>▶ CONTAINER-I-PDU</li> <li>▶ MULTIPLEXED-I-PDU</li> <li>▶ USER-DEFINED-I-PDU</li> <li>▶ DCM-I-PDU</li> <li>▶ GENERAL-PURPOSE-I-PDU</li> </ul> </li> <li>▶ SecOCRxPduSecuredArea/SecOCSecuredRxPduLength is set to SECURE-COMMUNICATION-PROPS/SECURED-AREA-LENGTH.</li> <li>▶ SecOCRxPduSecuredArea/SecOCSecuredRxPduOffset is set to SECURE-COMMUNICATION-PROPS/SECURED-AREA-OFFSET.</li> </ul>

Configuration parameters	Mapping description
	<p>► <b>SecOCRxAuthServiceConfigRef</b> is set to reference the <b>CsmJob</b> container that has been created for the <b>SEC-OC-CRYPTO-SERVICE-MAPPING</b> which the <b>PDU-TRIGGERING</b> of the <b>SECURED-I-PDU</b> is referencing.</p> <p>The content of the choice container <b>SecOCRxSecuredPduLayer</b> depends on whether the <b>SECURED-I-PDU</b> is a <i>cryptographic</i> PDU or not. If it is a <i>cryptographic</i> PDU, the content of the choice container is set to <b>SecOCRxSecuredPduCollection</b>, otherwise to <b>SecOCRxSecuredPdu</b>.</p> <p><b>SecOCRxSecuredPduCollection/SecOCRxAuthenticPdu/SecOCRxAuthenticPduRef</b> is set to reference the configuration container of the <i>payload</i> PDU in the <b>EcuC</b> module configuration.</p> <p><b>SecOCRxSecuredPduCollection/SecOCRxCryptographicPdu/SecOCRxCryptographicPduRef</b> is set to reference the configuration container of the <i>cryptographic</i> PDU in the <b>EcuC</b> module configuration.</p> <p><b>SecOCRxSecuredPduCollection/SecOCUseMessageLink/SecOCMessageLinkLen</b> is set to <b>SECURE-COMMUNICATION-PROPS/MESSAGE-LINK-LENGTH</b>, <b>SecOCRxSecuredPduCollection/SecOCUseMessageLink/SecOCMessageLinkPos</b> is set to <b>SECURE-COMMUNICATION-PROPS/MESSAGE-LINK-POSITION</b>.</p> <p><b>SecOCRxSecuredPduCollection/SecOCSecuredRxPduVerification</b> is set to <b>RX-SECURITY-VERIFICATION</b> of the <b>I-PDU-PORT</b> via which the <b>SECURED-I-PDU</b> is received.</p> <p><b>SecOCRxSecuredPdu/SecOCRxSecuredLayerPduRef</b> is set to reference the configuration container of the <b>SECURED-I-PDU</b> in the <b>EcuC</b> module configuration.</p> <p><b>SecOCRxSecuredPdu/SecOCSecuredRxPduVerification</b> is set to <b>RX-SECURITY-VERIFICATION</b> of the <b>I-PDU-PORT</b> via which the <b>SECURED-I-PDU</b> is received.</p> <p><b>SecOCRxSecuredPdu/SecOCAuthPduHeaderLength</b> and <b>SecOCRxSecuredPduCollection/SecOCRxAuthenticPdu/SecOCAuthPduHeaderLength</b> are configured in the same way as <b>SecOCTxSecuredPdu/SecOCAuthPduHeaderLength</b> and <b>SecOCTxSecuredPduCollection/SecOCTxAuthenticPdu/SecOCAuthPduHeaderLength</b>.</p>

### 3.4.43. SoAd

In its R19/11 release, AUTOSAR introduced a new communication and service model for Ethernet that supersedes the model that was present in AUTOSAR 4.4 and earlier releases. The AUTOSAR R19-11 schema allows to provide Ethernet communication and service data using either model. Both models are supported for the configuration of the `SoAd` module. It is, however, strongly discouraged to mix AUTOSAR 4.4 and R19-11 communication and service model concepts in a single Ecu Extract or System Description.

This chapter describes the configuration for both AUTOSAR 4.4 and R19-11 model, [5] and [6] provide in their "Ethernet specific description" chapters details on the differences of both models.

A `SOCKET-ADDRESS` conveys a parameter tuple (IP Address/Port Number/protocol). The protocol may either be `Tcp` or `Udp`. The IP Address and Port Number fields can either be defined statically, i.e. they provide a valid IPV4 or IPV6 address respectively a valid port number, or they can be defined dynamically. [Section 3.4.43.3, "Parameters derived from the local SOCKET-ADDRESS"](#) describes how the parameters are extracted out of a `SOCKET-ADDRESS` element.

Two `SOCKET-ADDRESS` elements are considered *distinct* if at least one of both has both IP Address and Port Number defined dynamically, or if both `SOCKET-ADDRESS` elements provide distinct values for the parameter fields they define statically.

In addition to that, two *remote* `SOCKET-ADDRESS` elements are also considered *distinct* if at least one of both has either IP Address or Port Number defined dynamically.

#### 3.4.43.1. SoAdSocketConnectionGroup

In the AUTOSAR R19-11 release, a `SoAdSocketConnectionGroup` container and one or more `SoAdSocketConnection` child containers are created in the following cases:

- ▶ One or more `STATIC-SOCKET-CONNECTIONS` exists which are contained in an `APPLICATION-ENDPOINT` which in turn is contained in a `SOCKET-ADDRESS` that refers to one of the `ETHERNET-COMMUNICATION-CONNECTOR` elements of the configured Ecu Instance. The name of the container is `CG<Name>`, where `<Name>` is the `SHORT-NAME` of the first `STATIC-SOCKET-CONNECTION`. The `SOCKET-ADDRESS` is used to configure the local IP address and port number as described in [Section 3.4.43.3, "Parameters derived from the local SOCKET-ADDRESS"](#). The role of the `SoAdSocketConnectionGroup` is *Server* if `STATIC-SOCKET-CONNECTION/TCP-ROLE` is `LISTEN`, otherwise it is *Client*.

One `SoAdSocketConnection` child container is created for each `STATIC-SOCKET-CONNECTION` that refers to a distinct `SOCKET-ADDRESS` via `STATIC-SOCKET-CONNECTION/REMOTE-ADDRESS`. Its name is the `SHORT-NAME` of the `STATIC-SOCKET-CONNECTION`. The `SoAdSocketRemoteIpAddress` child container is configured as (partly) static, see [Section 3.4.43.2, "Parameters derived from the remote SOCKET-ADDRESS"](#).

- ▶ The `APPLICATION-ENDPOINT` of a `SOCKET-ADDRESS` that refers to one of the `ETHERNET-COMMUNICATION-CONNECTOR` elements of the configured Ecu Instance is referenced by one or more `PROVID-`

ED-SERVICE-INSTANCE and/or CONSUMED-SERVICE-INSTANCE elements via PROVIDED-SERVICE-INSTANCE/LOCAL-UNICAST-ADDRESS or CONSUMED-SERVICE-INSTANCE/LOCAL-UNICAST-ADDRESS. The name of the container is CG<Name>\_<suffix>, where <Name> is the SHORT-NAME of the SOCKET-ADDRESS, and <suffix> is either a numerical index, or the SHORT-NAME of a remote SOCKET-ADDRESS. The referenced SOCKET-ADDRESS is used to configure the local IP address and port number as described in [Section 3.4.43.3, "Parameters derived from the local SOCKET-ADDRESS"](#). The role of the SOAD-SOCKET-CONNECTION-GROUP is **Server** if there is at least one PROVIDED-SERVICE-INSTANCE that references the APPLICATION-ENDPOINT of the SOCKET-ADDRESS, otherwise it is *Client*.

APPLICATION-ENDPOINT/MAX-NUMBER-OF-CONNECTIONS child SoAdSocketConnection containers are created if the parameter is defined. The container name is <Name>\_<suffix>, where <Name> is the SHORT-NAME of the SOCKET-ADDRESS, and <suffix> is a zero-based index. The SoAdSocketRemoteIpAddress is configured as dynamic, see [Section 3.4.43.2, "Parameters derived from the remote SOCKET-ADDRESS"](#).

Additional SoAdSocketConnection containers are created for each distinct SOCKET-ADDRESS which any of the PROVIDED-SERVICE-INSTANCE and/or CONSUMED-SERVICE-INSTANCE references via PROVIDED-SERVICE-INSTANCE/REMOTE-UNICAST-ADDRESS or CONSUMED-SERVICE-INSTANCE/REMOTE-UNICAST-ADDRESS which uses the same protocol (i.e. Tcp or Udp) as the local SOCKET-ADDRESS. The container name is <Name>\_<suffix>, where <suffix> is the SHORT-NAME of the remote SOCKET-ADDRESS. The SoAdSocketRemoteIpAddress is configured as (partly) static, see [Section 3.4.43.2, "Parameters derived from the remote SOCKET-ADDRESS"](#).

- The APPLICATION-ENDPOINT of a SOCKET-ADDRESS that refers to one of the ETHERNET-COMMUNICATION-CONNECTOR elements of the configured Ecu Instance is referenced by one or more CONSUMED-EVENT-GROUP elements via CONSUMED-EVENT-GROUP/EVENT-MULTICAST-ADDRESS. The name of the container is CG<Name>\_<suffix> where <Name> is the SHORT-NAME of the SOCKET-ADDRESS, and <suffix> is either a numerical index, or the SHORT-NAME of a remote SOCKET-ADDRESS. The SOCKET-ADDRESS is used to configure the local IP address and port number as described in [Section 3.4.43.3, "Parameters derived from the local SOCKET-ADDRESS"](#). Since the local address is supposed to be a multicast address which is not supported for Tcp, the role of the SoAdSocketConnectionGroup is not relevant in this case.

One SoAdSocketConnection container is created if the CONSUMED-SERVICE-INSTANCE of the CONSUMED-EVENT-GROUP refers to a Udp address via CONSUMED-SERVICE-INSTANCE/REMOTE-UNICAST-ADDRESS. The name of the container is the SHORT-NAME of that remote SOCKET-ADDRESS. The SoAdSocketRemoteIpAddress is configured as (partly) static, see [Section 3.4.43.2, "Parameters derived from the remote SOCKET-ADDRESS"](#). If APPLICATION-ENDPOINT/MAX-NUMBER-OF-CONNECTIONS is defined, then MAX-NUMBER-OF-CONNECTIONS child SoAdSocketConnection containers are created, named <Name>\_<suffix>, where <Name> is the SHORT-NAME of the SOCKET-ADDRESS, and <suffix> is a zero-based index. The SoAdSocketRemoteIpAddress is configured as dynamic, see [Section 3.4.43.2, "Parameters derived from the remote SOCKET-ADDRESS"](#).

If `APPLICATION-ENDPOINT/MAX-NUMBER-OF-CONNECTIONS` is not defined or zero, and the parent `CONSUMED-SERVICE-INSTANCE/REMOTE-UNICAST-ADDRESS` does not refer to a remote Udp address, one `SoAdSocketConnection` container is created, named `<Name>_0` in order to allow reception of PDUs. The `SoAdSocketRemoteIpAddress` is configured as dynamic, see [Section 3.4.43.2, “Parameters derived from the remote SOCKET-ADDRESS”](#).

In the AUTOSAR 4.4 and earlier releases, a `SoAdSocketConnectionGroup` and one or more `SoAdSocketConnection` child containers are created for the following cases:

- ▶ One or more `SOCKET-CONNECTION-BUNDLE` elements refer via `SOCKET-CONNECTION-BUNDLE/SERVER-PORT-REF` to a `SOCKET-ADDRESS` which in turn refers to one of the `ETHERNET-COMMUNICATION-CONNECTOR` elements of the configured Ecu Instance. In this case, the configured Ecu Instance acts as a server. The container name is `CG<Name>`, where `<Name>` is the `SHORT-NAME` of the `SOCKET-CONNECTION-BUNDLE`. The `SOCKET-ADDRESS` is used to configure the local IP address and port number as described in [Section 3.4.43.3, “Parameters derived from the local SOCKET-ADDRESS”](#).

For each distinct `SOCKET-ADDRESS` that is referenced by any `SOCKET-CONNECTION` of the `SOCKET-CONNECTION-BUNDLE`, one `SoAdSocketConnection` container is created. Its name is the `SHORT-LABEL` of the `SOCKET-CONNECTION`. If `SOCKET-CONNECTION/CLIENT-IP-ADDR-FROM-CONNECTION-REQUEST` is true, `SoAdSocketRemoteIpAddress/SoAdSocketRemoteIpAddress` is configured as ANY, otherwise the `SOCKET-ADDRESS` is taken to configure the remote IP address statically, see [Section 3.4.43.2, “Parameters derived from the remote SOCKET-ADDRESS”](#). If `SOCKET-CONNECTION/CLIENT-PORT-FROM-CONNECTION-REQUEST` is true, `SoAdSocketRemoteIpAddress/SoAdSocketRemotePort` is configured as 0, otherwise the `SOCKET-ADDRESS` is taken to configure the remote port statically, see [Section 3.4.43.2, “Parameters derived from the remote SOCKET-ADDRESS”](#).

- ▶ One or more `SOCKET-CONNECTION` elements refer via `SOCKET-CONNECTION/CLIENT-PORT-REF` to a `SOCKET-ADDRESS` which in turn refers to one of the `ETHERNET-COMMUNICATION-CONNECTOR` elements of the configured Ecu Instance. In this case, the configured Ecu Instance acts as a client. The container name is `CG<Name>`, where `<Name>` is the `SHORT-LABEL` of one of the `SOCKET-CONNECTION` elements. The `SOCKET-ADDRESS` is used to configure the local IP address and port number as described in [Section 3.4.43.3, “Parameters derived from the local SOCKET-ADDRESS”](#).

For each distinct `SOCKET-ADDRESS` that is referenced by any `SOCKET-CONNECTION-BUNDLE` of the `SOCKET-CONNECTION`, one `SoAdSocketConnection` container is created. Its name is the `SHORT-NAME` of the `SOCKET-CONNECTION-BUNDLE`. If `SOCKET-CONNECTION/RUNTIME-IP-ADDRESS-CONFIGURATION` is SD, `SoAdSocketRemoteIpAddress/SoAdSocketRemoteIpAddress` is configured as ANY, otherwise the `SOCKET-ADDRESS` is taken to configure the remote IP address statically, see [Section 3.4.43.2, “Parameters derived from the remote SOCKET-ADDRESS”](#). If `SOCKET-CONNECTION/RUNTIME-PORT-CONFIGURATION` is SD, `SoAdSocketRemoteIpAddress/SoAdSocketRemotePort` is configured as 0, otherwise the `SOCKET-ADDRESS` is taken to configure the remote port statically, see [Section 3.4.43.2, “Parameters derived from the remote SOCKET-ADDRESS”](#).

If any `SoAdPduRoute` or `SoAdSocketRoute` associated with the `SoAdSocketConnectionGroup` container or one of its `SoAdSocketConnection` child containers has `SoAdTxPduHeaderId` or `SoAdRxPduHeaderId` `PDU-ACTIVATION-ROUTING-GROUP` configured, then `SoAdPduHeaderEnable` is set to `true`, otherwise to `false`.

### 3.4.43.2. Parameters derived from the remote SOCKET-ADDRESS

The `SoAdSocketRemoteIpAddress` sub container of a `SoAdSocketConnection` can be configured as dynamic, as static, or as partly static, i.e. either IP address or port number is configured as dynamic whereas the other parameter is configured as static.

If the `SoAdSocketRemoteIpAddress` is configured as dynamic, `SoAdSocketRemoteIpAddress/SoAdSocketRemoteIpAddress` is configured as `ANY` and `SoAdSocketRemoteIpAddress/SoAdSocketRemotePort` is configured as `0`. At runtime, the socket can be reached via any IP Address/Port number combination provided the protocol, i.e. `Tcp/Udp` is matching the protocol configured in the parent `SoAdSocketConnectionGroup`.

If the `SoAdSocketRemoteIpAddress` is configured as static or partly static, either the remote IP Address, or the remote Port, or both are determined at configuration time by the properties of the remote `SOCKET-ADDRESS`. If it contains an `APPLICATION-ENDPOINT` which in turn contains a `TCP-TP` or `UDP-TP` element providing a valid `PORT-NUMBER` value and `DYNAMICALLY-ASSIGNED` is either not defined or set to `false`, then `PORT-NUMBER` is used to configure `SoAdSocketRemoteIpAddress/SoAdSocketRemotePort`. If the `APPLICATION-ENDPOINT` of the remote `SOCKET-ADDRESS` refers to a `NETWORK-ENDPOINT` which either contains an `IPV-4-CONFIGURATION` or an `IPV-6-CONFIGURATION`, which contains a valid IP Address in `IPV-4-CONFIGURATION/IPV-4-ADDRESS` or in `IPV-6-CONFIGURATION/IPV-6-ADDRESS`, then this IP address is used to configure `SoAdSocketRemoteIpAddress/SoAdSocketRemoteIpAddress`.

### 3.4.43.3. Parameters derived from the local SOCKET-ADDRESS

`SoAdSocketLocalAddressRef` is set up to refer to the `TcpIpLocalAddr` container in the `TcpIp` module that has been created for the local `SOCKET-ADDRESS`.

If the local `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` which in turn contains a `TCP-TP` or `UDP-TP` element providing a valid `PORT-NUMBER` value and `DYNAMICALLY-ASSIGNED` is either not defined or set to `false`, then `PORT-NUMBER` is used to configure `SoAdSocketLocalPort`.

The `SoAdSocketProtocol` sub container is configured depending to the child element of the `TP-CONFIGURATION` of the `APPLICATION-ENDPOINT` in the local `SOCKET-ADDRESS`. `SoAdSocketProtocol` is configured as `SoAdSocketUdp` if the child is a `UDP-TP` element and configured as `SoAdSocketTcp` if the child is a `TCP-TP` element.



If the local `SOCKET-ADDRESS` has one or more `PROVIDED-SERVICE-INSTANCE` associated, i.e. the `APPLICATION-ENDPOINT` of the `SOCKET-ADDRESS` contains the `PROVIDED-SERVICE-INSTANCE` elements, or the `PROVIDED-SERVICE-INSTANCE` elements reference that `APPLICATION-ENDPOINT` via `PROVIDED-SERVICE-INSTANCE/LOCAL-UNICAST-ADDRESS`, and the `PROVIDED-SERVICE-INSTANCE` elements contain a valid and consistent `PRIORITY` value, then that value is used to configure `SoAdSocketFramePriority`.

If the local `SOCKET-ADDRESS` has one or more `CONSUMED-EVENT-GROUP` associated, i.e. the `APPLICATION-ENDPOINT` of the `SOCKET-ADDRESS` contains `CONSUMED-SERVICE-INSTANCE` elements which in turn contain the `CONSUMED-EVENT-GROUP` elements, or the `CONSUMED-EVENT-GROUPS` reference that `APPLICATION-ENDPOINT` via `CONSUMED-EVENT-GROUP/EVENT-MULTICAST-ADDRESS`, and the `CONSUMED-EVENT-GROUP` elements contain a valid and consistent `PRIORITY` value, then that value is used to configure `SoAdSocketFramePriority`.

If `SoAdSocketFramePriority` cannot be derived from `PROVIDED-SERVICE-INSTANCE` or `CONSUMED-EVENT-GROUP` elements, the `APPLICATION-ENDPOINT/PRIORITY` is used to configure `SoAdSocketFramePriority`. If `APPLICATION-ENDPOINT/PRIORITY` is also not present, `NETWORK-ENDPOINT/PRIORITY` of the `NETWORK-ENDPOINT` referenced via `APPLICATION-ENDPOINT/NETWORK-ENDPOINT-REF` is used to configure `SoAdSocketFramePriority`.

If the local `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` which in turn contains a `TCP-TP` element, the following parameters are configured in `SoAdSocketTcp`:

- ▶ `SoAdSocketTcpNoDelay` from the negated value of `NAGLES-ALGORITHM`
- ▶ `SoAdSocketTcpKeepAliveProbesMax` from `KEEP-ALIVE-PROBE-MAX`
- ▶ `SoAdSocketTcpKeepAliveInterval` from `KEEP-ALIVE-INTERVAL`
- ▶ `SoAdSocketTcpKeepAliveTime` from `KEEP-ALIVE-TIME`

If any of `SoAdSocketTcp/SoAdSocketTcpKeepAliveProbesMax`, `SoAdSocketTcp/SoAdSocketTcpKeepAliveInterval`, or `SoAdSocketTcp/SoAdSocketTcpKeepAliveTime` is configured or if the local `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` which in turn contains a `TCP-TP` element providing a `KEEP-ALIVES` field which is set to `true`, then `SoAdSocketTcp/SoAdSocketTcpKeepAlive` is also configured as `true`.

#### 3.4.43.4. Local `SOCKET-ADDRESS` elements configured by `Sd`

A local `SOCKET-ADDRESS` of the configured `ECU-INSTANCE` is configured by the `Sd` at runtime if the configured `ECU-INSTANCE` sends or receives one or more PDUs via the `SOCKET-ADDRESS` and one of the following conditions applies:

- ▶ The `SOCKET-ADDRESS` is referenced by an AUTOSAR 4.4 `SOCKET-CONNECTION` via `CLIENT-PORT-REF` and the `SOCKET-CONNECTION` has its `RUNTIME-IP-ADDRESS-CONFIGURATION` set to `SD`.

- The `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT`, which in turn contains an `AUTOSAR R19-11 CONSUMED-EVENT-GROUP` of the configured `ECU-INSTANCE` references via `EVENT-MULTICAST-ADDRESS`.

### 3.4.43.5. Parameters derived from other entities

If the local `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` which in turn contains a `TCP-TP` element, then `SoAdSocketTcp/SoAdSocketTcpInitiate` is configured depending on the role of the `SoAdSocketConnectionGroup`, see [Section 3.4.43.1, “SoAdSocketConnectionGroup”](#). If the role is *Client*, then `SoAdSocketTcp/SoAdSocketTcpInitiate` is configured as `true`. `SoAdSocketTcp/SoAdSocketTcpInitiate` is configured as `false` if the `SoAdSocketConnectionGroup` container contains only one `SoAdSocketConnection` child container and its role is *Server*.

If the local `SOCKET-ADDRESS` contains an `APPLICATION-ENDPOINT` which in turn contains a `UDP-TP` element, then `SoAdSocketUdp/SoAdSocketUdpTriggerTimeout` and `SoAdSocketUdp/SoAdSocketnPduUdpTxBufferMin` are configured depending on the entity for which the `SoAdSocketConnectionGroup` was created.

If the `SoAdSocketConnectionGroup` was created for a set of `SOCKET-CONNECTION-BUNDLE` elements, or for a set of `SOCKET-CONNECTION` elements, then all `SOCKET-CONNECTIONS` respectively all `SOCKET-CONNECTION` elements contained in all `SOCKET-CONNECTION-BUNDLE` elements are examined. If one of the `SOCKET-CONNECTION` elements has a valid `PDU-COLLECTION-TIMEOUT` parameter defined, then this value is taken to configure `SoAdSocketUdp/SoAdSocketUdpTriggerTimeout`. If one of the `SOCKET-CONNECTION` elements has a valid `PDU-COLLECTION-MAX-BUFFER-SIZE` parameter defined, then this value is taken to configure `SoAdSocketUdp/SoAdSocketnPduUdpTxBufferMin`.

If the `SoAdSocketConnectionGroup` was created for a `STATIC-SOCKET-CONNECTION` or for a `SOCKET-ADDRESS` referenced by one or more `PROVIDED-SERVICE-INSTANCE` and/or `CONSUMED-SERVICE-INSTANCE` elements, then `SoAdSocketUdp/SoAdSocketUdpTriggerTimeout` and `SoAdSocketUdp/SoAdSocketnPduUdpTxBufferMin` are configured depending on the local `SOCKET-ADDRESS`: If the local `SOCKET-ADDRESS` contains a valid `PDU-COLLECTION-TIMEOUT` value, then this value is taken to configure `SoAdSocketUdp/SoAdSocketUdpTriggerTimeout`. If the local `SOCKET-ADDRESS` contains a valid `PDU-COLLECTION-MAX-BUFFER-SIZE` value, then this value is taken to configure `SoAdSocketUdp/SoAdSocketnPduUdpTxBufferMin`.

`SoAdSocketUdp/SoAdSocketUdpRetryEnabled` is configured depending on whether there is any PDU sent/received via `Udp` that has a length such that  $(IP\_HEADER\_LEN + UDP\_HEADER\_LEN + PDU\_LEN) > MTU$ . In that case, `SoAdSocketUdp/SoAdSocketUdpRetryEnabled` is configured as `true`, otherwise it is configured as `false`. `IP_HEADER_LEN` in the above formula is 20 for `IPV4` and 40 for `IPV6`, and `UDP_HEADER_LEN` is 8. The length of a PDU is calculated as described in [Section 3.3.7, “PDU length calculation”](#).

If either the local or the remote `SOCKET-ADDRESS` refers to a `TLS-CRYPTO-SERVICE-MAPPING` which in turn contains a `TLS-CRYPTO-CIPHER-SUITE` for which a `TlsConnection` container was created in the

configuration of the `Tls` module, then `SoAdSocketConnection/SoAdTlsConnectionRef` is configured to refer to that `TlsConnection` container. [Section 3.4.47, “Tls”](#) describes the configuration of `TlsConnection` containers.

### 3.4.43.6. SoAdPduRoute and SoAdSocketRoute containers

Since AUTOSAR R19-11 `SO-CON-I-PDU-IDENTIFIER` replaces `SOCKET-CONNECTION-IPDU-IDENTIFIER`, and `PDU-ACTIVATION-ROUTING-GROUP` replaces `SO-AD-ROUTING-GROUP`. Both old and new entities convey roughly the same information, however, the relationship between them and the relationship to the entities for which `SoAdSocketConnectionGroup` and `SoAdSocketConnection` containers are created are modeled differently.

A `SO-CON-I-PDU-IDENTIFIER` belongs to a `PDU-ACTIVATION-ROUTING-GROUP` if the `PDU-ACTIVATION-ROUTING-GROUP` references the `SO-CON-I-PDU-IDENTIFIER` either via `I-PDU-IDENTIFIER-TCP-REFS` or via `I-PDU-IDENTIFIER-UDP-REFS`. The role via which the `SO-CON-I-PDU-IDENTIFIER` is referenced also defines the protocol used for sending/receiving.

A `SOCKET-CONNECTION-IPDU-IDENTIFIER` belongs to a `SO-AD-ROUTING-GROUP` if the `SOCKET-CONNECTION-IPDU-IDENTIFIER` references the `SO-AD-ROUTING-GROUP`.

A `SO-CON-I-PDU-IDENTIFIER` belongs to a `SoAdSocketConnection` container if the `SoAdSocketConnection` was created for a `STATIC-SOCKET-CONNECTION` and the `STATIC-SOCKET-CONNECTION` references the `SO-CON-I-PDU-IDENTIFIER`.

A `SO-CON-I-PDU-IDENTIFIER` belongs to a `SoAdSocketConnectionGroup` or one of its `SoAdSocketConnection` child containers if the `SoAdSocketConnectionGroup` was created for a local `SOCKET-ADDRESS` element that contains an `APPLICATION-ENDPOINT` which in turn is referenced by one or more `PROVIDED-SERVICE-INSTANCE` and/or `CONSUMED-SERVICE-INSTANCE` elements and one of the following condition holds:

- ▶ The `PROVIDED-SERVICE-INSTANCE/CONSUMED-SERVICE-INSTANCE` contains a `PDU-ACTIVATION-ROUTING-GROUP` which in turn references the `SO-CON-I-PDU-IDENTIFIER`.
- ▶ The `PROVIDED-SERVICE-INSTANCE` contains an `EVENT-HANDLER`, which aggregates a `PDU-ACTIVATION-ROUTING-GROUP` which in turn references the `SO-CON-I-PDU-IDENTIFIER`.
- ▶ The `CONSUMED-SERVICE-INSTANCE` contains a `CONSUMED-EVENT-GROUP`, which aggregates a `PDU-ACTIVATION-ROUTING-GROUP` which in turn references the `SO-CON-I-PDU-IDENTIFIER`.

In all cases the protocol of the `SOCKET-ADDRESS` and the role of the reference of the `PDU-ACTIVATION-ROUTING-GROUP` must match, i.e. if the `SOCKET-ADDRESS` represents a `Udp` address, the `SO-CON-I-PDU-IDENTIFIER` must be referenced via `PDU-ACTIVATION-ROUTING-GROUP/I-PDU-IDENTIFIER-UDP-REFS`, if the `SOCKET-ADDRESS` represents a `Tcp` address, the `SO-CON-I-PDU-IDENTIFIER` must be referenced via `PDU-ACTIVATION-ROUTING-GROUP/I-PDU-IDENTIFIER-TCP-REFS`. If

the `PROVIDED-SERVICE-INSTANCE/CONSUMED-SERVICE-INSTANCE` references a remote IP address that matches the protocol via `REMOTE-UNICAST-ADDRESS`, the `SO-CON-I-PDU-IDENTIFIER` belongs to the `SoAdSocketConnection` that was created for that remote address, otherwise the `SO-CON-I-PDU-IDENTIFIER` belongs to the `SoAdSocketConnectionGroup`.

A `SO-CON-I-PDU-IDENTIFIER` belongs to a `SoAdSocketConnectionGroup` or one of its `SoAdSocketConnection` child containers if the `SoAdSocketConnectionGroup` was created for a local `SOCKET-ADDRESS` element that contains an `APPLICATION-ENDPOINT` which in turn is referenced by one or more `CONSUMED-SERVICE-INSTANCE` elements via `CONSUMED-SERVICE-INSTANCE/EVENT-MULTICAST-ADDRESS` and the following conditions hold:

- ▶ The `CONSUMED-SERVICE-INSTANCE` contains a `PDU-ACTIVATION-ROUTING-GROUP` which in turn references the `SO-CON-I-PDU-IDENTIFIER` via `PDU-ACTIVATION-ROUTING-GROUP/I-PDU-IDENTIFIER-UDP-REFS`.
- ▶ The `SOCKET-ADDRESS` represents a Udp multicast address.

If the `CONSUMED-SERVICE-INSTANCE` references a remote `SOCKET-ADDRESS` that represents a Udp address the `SO-CON-I-PDU-IDENTIFIER` belongs to the `SoAdSocketConnection` that was created for that remote address, otherwise the `SO-CON-I-PDU-IDENTIFIER` belongs to the `SoAdSocketConnectionGroup`.

A `SOCKET-CONNECTION-IPDU-IDENTIFIER` belongs to a `SoAdSocketConnectionGroup` container if the `SoAdSocketConnectionGroup` was created for a `SocketConnectionBundle` and the `SocketConnectionBundle` contains the `SOCKET-CONNECTION-IPDU-IDENTIFIER`.

A `SOCKET-CONNECTION-IPDU-IDENTIFIER` belongs to a `SoAdSocketConnectionGroup` container if the `SoAdSocketConnectionGroup` was created for a list of `SOCKET-CONNECTION` elements and all of the `SOCKET-CONNECTION` elements contain a `SOCKET-CONNECTION-IPDU-IDENTIFIER` with the identical properties.

A `SOCKET-CONNECTION-IPDU-IDENTIFIER` belongs to a `SoAdSocketConnection` container if the `SoAdSocketConnection` container was created for a `SOCKET-CONNECTION` and the `SOCKET-CONNECTION` contains the `SOCKET-CONNECTION-IPDU-IDENTIFIER`.

The following sections refer to `SO-CON-I-PDU-IDENTIFIER` and `SOCKET-CONNECTION-IPDU-IDENTIFIER` as `SCII`, and to `PDU-ACTIVATION-ROUTING-GROUP` and `SO-AD-ROUTING-GROUP` as `ROUTING-GROUP` if statements apply to both types at the same time.

An `SCII` is related to a `PDU-TRIGGERING` if it references the `PDU-TRIGGERING`.

A `PDU` is related to an `SCII` if the `SCII` references a `PDU-TRIGGERING` which in turn references the `PDU`, and the imported `ECU-INSTANCE` sends, receives, or routes that `PDU`. For further information about routing see [Section 3.3.6, “PDU routing”](#).

### 3.4.43.7. SoAdPduRoute

All SCII elements associated with sent PDUs are grouped according to their sent PDU. One `SoAdPduRoute` sub container is created for each distinct sent PDU. The container name is `PR<PREFIX><name><INSTSUFFIX>`, where `<name>` is the `SHORT-NAME` of the PDU.

`SoAdTxPduRef` references the corresponding PDU container in the `EcuC` module.

`SoAdTxUpperLayerType` is set to `TP` in the following cases:

- ▶ An `ETH-TP-CONNECTION` refers to the `PDU-TRIGGERING` that references the transmitted PDU and that belongs to the `PHYSICAL-CHANNEL` on which the PDU is transmitted.
- ▶ For `GENERAL-PURPOSE-PDU` elements with `CATEGORY` set to `DoIP`.
- ▶ For `GENERAL-PURPOSE-PDU` elements with `CATEGORY` set to `Dlt_TP`.
- ▶ For `DCM-I-PDU` elements unless they are referenced by a `PDU-TRIGGERING` which in turn is referenced via `DIAGNOSTIC-CONNECTION/PERIODIC-RESPONSE-UUDT-REF`.

`SoAdTxUpperLayerType` is set to `IF` in the following cases:

- ▶ For `GENERAL-PURPOSE-PDU` elements with `CATEGORY` set to `Dlt_IF`.

In all other cases, `SoAdTxUpperLayerType` is set to `IF` as a default.

`SoAdTxPduCollectionSemantics` is configured by collecting the `PDU-COLLECTION-SEMANTICS` values of all SCII elements that belong to this `SoAdPduRoute`. If that collection yields exactly one distinct value, that value is used to configure `SoAdTxPduCollectionSemantics`:

- ▶ `SOAD_COLLECT_LAST_IS_BEST` is configured for `LAST-IS-BEST`
- ▶ `SOAD_COLLECT_QUEUED` is configured for `QUEUED`

### 3.4.43.8. SoAdPduRoute/SoAdPduRouteDest

One `SoAdPduRouteDest` child container is created for each SCII of a sent PDU for which the parent `SoAdPduRoute` was created unless the type of the received PDU is listed in [Section 3.4.43.12, “PDU types without transmission support on SoAdSocketConnectionGroup level”](#). If the SCII is associated with a `SoAdSocketConnectionGroup` container, see [Section 3.4.43.6, “SoAdPduRoute and SoAdSocketRoute containers”](#), only one `SoAdPduRouteDest` container is created. The container name is `<name>_<headerId>`, where `<name>` is the name of the container that represents the related `SoAdSocketConnectionGroup` or `SoAdSocketConnection` container and `<headerId>` is the `HEADER-ID` of the SCII. If no `HEADER-ID` is available the container name is `<name>`.

`SoAdTxPduHeaderId` is set to `HEADER-ID` of the SCII referencing the sent PDU.

`SoAdTxUdpTriggerMode` is set depending on `PDU-COLLECTION-TRIGGER` of the SCII referencing the sent PDU: `TRIGGER_ALWAYS` for `ALWAYS` and `TRIGGER_NEVER` for `NEVER`.

`SoAdTxUdpTriggerTimeout` is set to `PDU-COLLECTION-PDU-TIMEOUT` of the `SCII` which references the sent PDU.

`SoAdTxSocketConnOrSocketConnBundleRef` references the `SoAdSocketConnectionGroup` or `SoAdSocketConnection` as described in [Section 3.4.43.13, “SocketConnOrSocketConnBundleRef”](#).

`SoAdTxRoutingGroupRef` references all `ROUTING-GROUP` containers created for the `ROUTING-GROUP` elements of the `SCII`.

### 3.4.43.9. SoAdSocketRoute

All `SCII` elements associated with received PDUs are grouped according to their `HEADER-ID` field and the `SoAdSocketConnection/SoAdSocketConnectionGroup` container they are associated with. One container is created for each group unless the type of the received PDU is listed in [Section 3.4.43.12, “PDU types without transmission support on SoAdSocketConnectionGroup level”](#). For more information about received or routed PDUs, see [Section 3.3.6, “PDU routing”](#) and [Section 3.4.46, “TcpIp”](#). The container name is `SR<PREFIX><name>_<headerId>` where `<name>` is the name of the `SoAdSocketConnectionGroup` or `SoAdSocketConnection` of the group, and `<headerId>` is the value of `HEADER-ID`. If no `HEADER-ID` is available the container name is `SR<PREFIX><name>`.

`SoAdRxPduHeaderId` is set to the `HEADER-ID` of the group.

`SoAdRxSocketConnOrSocketConnBundleRef` references the `SoAdSocketConnectionGroup` or `SoAdSocketConnection` as described in [Section 3.4.43.13, “SocketConnOrSocketConnBundleRef”](#).

### 3.4.43.10. SoAdSocketRoute/SoAdSocketRouteDest

One `SoAdSocketRouteDest` child container is created for each `SCII` that yields the `HEADER-ID` value of the parent `SoAdSocketRoute` container and which is associated with the `SoAdSocketConnection` or `SoAdSocketConnectionGroup` the parent `SoAdSocketRoute` container has been created for. The container name is `<PREFIX><name><INSTSUFFIX>`, where `<name>` is the `SHORT-NAME` of the PDU.

`SoAdRxPduRef` references the corresponding PDU container in the `EcuC` module.

`SoAdRxUpperLayerType` is configured in the same way as `SoAdTxUpperLayerType`.

If the `SoAdSocketLocalAddressRef` parameter of the associated `SoAdSocketConnectionGroup` refers to a multicast address, `SoAdRxRoutingGroupRef` references all `SoAdRoutingGroup` containers that have been created for `ROUTING-GROUP` elements that have `EVENT-GROUP-CONTROL-TYPE` set to `ACTIVATION-MULTICAST` and that are associated with the `SCII`.

If the `SoAdSocketLocalAddressRef` parameter of the associated `SoAdSocketConnectionGroup` refers to a unicast address, `SoAdRxRoutingGroupRef` references all `SoAdRoutingGroup` containers that have been created for `ROUTING-GROUP` elements that have `EVENT-GROUP-CONTROL-TYPE` set to a value other than `ACTIVATION-MULTICAST` and that are associated with the `SCII`.

### 3.4.43.11. SoAdRoutingGroup

For each unique ROUTING-GROUP associated with an SCII for which a SoAdPduRoute and/or a SoAdSocketRoute container has been created, a SoAdRoutingGroup container is created. The container name is RG<PREFIX><name> where <name> is the SHORT-NAME of the ROUTING-GROUP.

SoAdRoutingGroupTxTriggerable is set to true, if EVENT-GROUP-CONTROL-TYPE is set to ACTIVATION-AND-TRIGGER-UNICAST or TRIGGER-UNICAST.

### 3.4.43.12. PDU types without transmission support on SoAdSocketConnection-Group level

For the following PDU types dedicated SoAdPduRoute and SoAdSocketRoute containers are created for every SoAdSocketConnection child container of a SoAdSocketConnectionGroup container, even if the SCII is associated with the SoAdSocketConnectionGroup container:

PDU type	Category	Protocol
GENERAL-PURPOSE-PDU	DoIP	Tcp
GENERAL-PURPOSE-PDU	DoIP	Udp

### 3.4.43.13. SocketConnOrSocketConnBundleRef

SoAdPduRoute and SoAdSocketRoute containers reference SoAdSocketConnectionGroup or SoAdSocketConnection containers via the parameters SoAdTxSocketConnOrSocketConnBundleRef and SoAdRxSocketConnOrSocketConnBundleRef.

If a SoAdPduRoute/SoAdSocketRoute references a SoAdSocketConnectionGroup, the PDUs related to the SoAdPduRoute/SoAdSocketRoute can be sent/received via any of the SoAdSocketConnection elements of the SoAdSocketConnectionGroup at runtime.

Whether or not a SoAdPduRoute/SoAdSocketRoute references a SoAdSocketConnectionGroup or a SoAdSocketConnection depends on the requirements of the ROUTING-GROUP elements associated with the PDUs and on the properties of the SCII which specifies HEADER-ID and PDU.

An SCII may be associated with one or more ROUTING-GROUP elements. They are also referenced by the following entities that are related to service oriented communication:

- ▶ PROVIDED-SERVICE-INSTANCE
- ▶ CONSUMED-SERVICE-INSTANCE
- ▶ CONSUMED-EVENT-GROUP
- ▶ EVENT-HANDLER



If all ROUTING-GROUP elements of a SOCKET-CONNECTION-IPDU-IDENTIFIER are referenced from entities of one single type, then that type determines whether the SoAdPduRoute/SoAdSocketRoute references a SoAdSocketConnectionGroup or a SoAdSocketConnection. If that type is PROVIDED-SERVICE-INSTANCE, the SoAdPduRoute/SoAdSocketRoute references a SoAdSocketConnection. For all other types, the SoAdPduRoute/SoAdSocketRoute references a SoAdSocketConnectionGroup.

If none or more than one type was found, the SoAdPduRoute/SoAdSocketRoute references the SoAdSocketConnection or SoAdSocketConnectionGroup container of the SCII, according to [Section 3.4.43.6, "SoAdPduRoute and SoAdSocketRoute containers"](#).

### 3.4.44. SomeIpTp

Configuration parameters	Mapping description
SomeIpTp/SomeIpTpChannel	<p>For every SOMEIP-TP-CONNECTION element that refers via TP-SDU-REF to a PDU which the imported ECU-INSTANCE sends or receives, a SomeIpTpChannel container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>If SOMEIP-TP-CONNECTION refers to a SOMEIP-TP-CHANNEL which contains a valid SEPARATION-TIME, that value is used to configure SomeIpTpNPduSeparationTime. Otherwise, the SEPARATION-TIME value of the SOMEIP-TP-CONNECTION is used for that purpose.</p> <p>If SOMEIP-TP-CONNECTION refers to a SOMEIP-TP-CHANNEL which contains a valid RX-TIMEOUT-TIME, that value is used to configure SomeIpTpRxTimeoutTime.</p>
SomeIpTp/SomeIpTpChannel/SomeIpTpRxNSdu	<p>If the SOMEIP-TP-CONNECTION refers to a PDU that the imported ECU-INSTANCE receives, one SomeIpTpRxNSdu sub container is created. Its name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the PDU.</p> <p>SomeIpTpRxSduRef is set to reference the configuration container of the PDU in the EcuC module configuration.</p>
SomeIpTp/SomeIpTpChannel/SomeIpTpRxNSdu/SomeIpTpRxNPdu	<p>One SomeIpTpRxNPdu is created if the SOMEIP-TP-CONNECTION refers via TRANSPORT-PDU-REF to a PDU for which the following conditions hold:</p> <ul style="list-style-type: none"> <li>▶ The referenced PDU is of type GENERAL-PURPOSE-I-PDU</li> <li>▶ The CATEGORY field of the PDU is set to SOMEIP_SEGMENTED_IPDU</li> <li>▶ The imported ECU-INSTANCE receives the PDU</li> </ul>



Configuration parameters	Mapping description
	<code>SomeIpTpRxNPduRef</code> is set to reference the configuration container of the PDU in the <code>EcuC</code> module configuration.
<code>SomeIpTp/SomeIpTpChannel/SomeIpTpTxNSdu</code>	<p>If the <code>SOMEIP-TP-CONNECTION</code> refers to a PDU that the imported <code>ECU-INSTANCE</code> sends, one <code>SomeIpTpTxNSdu</code> sub container is created. Its name is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;</code>, where <code>&lt;name&gt;</code> is the <code>SHORT-NAME</code> of the PDU.</p> <p><code>SomeIpTpTxNSduRef</code> is set to reference the configuration container of the PDU in the <code>EcuC</code> module configuration.</p>
<code>SomeIpTp/SomeIpTpChannel/SomeIpTpRxNSdu/SomeIpTpRxNPdu</code>	<p>One <code>SomeIpTpRxNPdu</code> is created if the <code>SOMEIP-TP-CONNECTION</code> refers via <code>TRANSPORT-PDU-REF</code> to a PDU for which the following conditions hold:</p> <ul style="list-style-type: none"> <li>▶ The referenced PDU is of type <code>GENERAL-PURPOSE-I-PDU</code></li> <li>▶ The <code>CATEGORY</code> field of the PDU is set to <code>SOMEIP_SEGMENTED_IPDU</code></li> <li>▶ The imported <code>ECU-INSTANCE</code> sends the PDU</li> </ul> <p><code>SomeIpTpTxNPduRef</code> is set to reference the configuration container of the PDU in the <code>EcuC</code> module configuration.</p>

### 3.4.45. StbM

Configuration parameters	Mapping description
<code>StbMSynchronizedTimeBase</code>	<p>One <code>StbMSynchronizedTimeBase</code> container is created for each <code>GLOBAL-TIME-DOMAIN</code> that fulfills the following conditions:</p> <ul style="list-style-type: none"> <li>▶ The imported <code>ECU-INSTANCE</code> belongs to the <code>GLOBAL-TIME-DOMAIN</code>.</li> <li>▶ The <code>GLOBAL-TIME-DOMAIN</code> does not belong to a parent <code>GLOBAL-TIME-DOMAIN</code> to which the imported <code>ECU-INSTANCE</code> belongs as well.</li> </ul> <p>The imported <code>ECU-INSTANCE</code> belongs to a <code>GLOBAL-TIME-DOMAIN</code> in the following cases:</p> <ul style="list-style-type: none"> <li>▶ The <code>GLOBAL-TIME-MASTER</code> of the <code>GLOBAL-TIME-DOMAIN</code> references a <code>COMMUNICATION-CONNECTOR</code> of the imported <code>ECU-INSTANCE</code></li> <li>▶ One of the <code>GLOBAL-TIME-SLAVE</code> elements of the <code>GLOBAL-TIME-DOMAIN</code> references a <code>COMMUNICATION-CONNECTOR</code> of the imported <code>ECU-INSTANCE</code></li> </ul>

Configuration parameters	Mapping description
	<p>► The GLOBAL-TIME-DOMAIN is considered a parent GLOBAL-TIME-DOMAIN of a GLOBAL-TIME-DOMAIN of the imported ECU-INSTANCE</p> <p>A GLOBAL-TIME-DOMAIN gtp is considered to be the parent GLOBAL-TIME-DOMAIN of a GLOBAL-TIME-DOMAIN gts in the following cases:</p> <ul style="list-style-type: none"> <li>► gts contains a GLOBAL-TIME-GATEWAY which references a GLOBAL-TIME-MASTER contained in gts and a GLOBAL-TIME-SLAVE contained in gtp.</li> <li>► gts either contains a GLOBAL-TIME-MASTER or GLOBAL-TIME-SLAVE, and gtp neither contains a GLOBAL-TIME-MASTER nor any GLOBAL-TIME-SLAVE, but references gts in SUB-DOMAIN-REFS.</li> </ul> <p>The name of an imported StbMSynchronizedTimeBase container is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the GLOBAL-TIME-DOMAIN.</p> <p>StbMSynchronizedTimeBaseIdentifier is set to DOMAIN-ID.</p> <p>StbMSyncLossTimeout is set to SYNC-LOSS-TIMEOUT.</p> <p>StbMTimeLeapFutureThreshold is set to TIME-LEAP-FUTURE-THRESHOLD.</p> <p>StbMTimeLeapPastThreshold is set to TIME-LEAP-PAST-THRESHOLD.</p> <p>StbMClearTimeleapCount is set to TIME-LEAP-HEALING-COUNTER.</p> <p>For the configuration of StbMIsSystemWideGlobalTimeMaster, all GLOBAL-TIME-MASTER elements of the GLOBAL-TIME-DOMAIN are retrieved. If the GLOBAL-TIME-DOMAIN is <i>abstract</i>, i.e. it does not reference a GLOBAL-TIME-MASTER and no GLOBAL-TIME-SLAVE, the GLOBAL-TIME-MASTER elements that reference a COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE of all GLOBAL-TIME-DOMAIN elements that are referenced via SUB-DOMAIN-REF are collected.</p> <p>If the GLOBAL-TIME-DOMAIN is not <i>abstract</i>, the GLOBAL-TIME-MASTER in GLOBAL-TIME-DOMAIN/MASTER is collected if it references a COMMUNICATION-CONNECTOR of the imported ECU-INSTANCE.</p> <p>If no GLOBAL-TIME-MASTER was retrieved in the previous step, false is configured. If all retrieved GLOBAL-TIME-MASTER elements yield the same IS-</p>

Configuration parameters	Mapping description
	<p>SYSTEM-WIDE-GLOBAL-TIME-MASTER value, that value is configured. Otherwise, nothing is configured.</p> <p>If OFFSET-TIME-DOMAIN-REF refers to a valid GLOBAL-TIME-DOMAIN, StbMOffsetTimeBase is configured to reference the StbMSynchronizedTimeBase container that has been created for that GLOBAL-TIME-DOMAIN.</p>
StbMSynchronizedTimeBase/StbMTimeCorrection	<p>If the GLOBAL-TIME-DOMAIN contains a GLOBAL-TIME-CORRECTION-PROPS, then a StbMTimeCorrection container is created, and following parameters are set:</p> <p>StbMOffsetCorrectionAdaptionInterval is set to OFFSET-CORRECTION-ADAPTION-INTERVAL.</p> <p>StbMOffsetCorrectionJumpThreshold is set to OFFSET-CORRECTION-JUMP-THRESHOLD.</p> <p>StbMRateCorrectionMeasurementDuration is set to RATE-CORRECTION-MEASUREMENT-DURATION.</p> <p>StbMRateCorrectionsPerMeasurementDuration is set to RATE-CORRECTIONS-PER-MEASUREMENT-DURATION.</p>

### 3.4.46. Tcplp

Configuration parameters	Mapping description
TcpIpConfig/TcpIpCtrl	<p>For every <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER of the imported ECU-INSTANCE (see <a href="#">Section 3.4.17, “EthIf”</a>), a TcpIpCtrl container is created. The container name is &lt;PREFIX&gt;&lt;name_cc&gt;_&lt;name_chn&gt;, where &lt;name_cc&gt; is the SHORT-NAME of the ETHERNET-COMMUNICATION-CONTROLLER and &lt;name_chn&gt; is the SHORT-NAME of the ETHERNET-PHYSICAL-CHANNEL.</p> <p>TcpIpEthIfCtrlRef references the container created for this <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER in the EthIf module.</p> <p>TcpIpIpFramePrioDefault is set to the value of DEFAULT-PRIORITY of the VLAN-MEMBERSHIP belonging to the ETHERNET-PHYSICAL-CHANNEL of the ETHERNET-COMMUNICATION-CONTROLLER. A VLAN-MEMBERSHIP belongs to an ETHERNET-PHYSICAL-CHANNEL if it references this ETHER-</p>

Configuration parameters	Mapping description
	<p>NET-PHYSICAL-CHANNEL and is aggregated by a COUPLING-PORT of the ETHERNET-COMMUNICATION-CONTROLLER. If DEFAULT-PRIORITY is not available, TcpIpIpFramePrioDefault is set to 0</p> <p>For every ETHERNET-COMMUNICATION-CONNECTOR associated with a NETWORK-ENDPOINT with valid IPV-4-CONFIGURATION or IPV-6-CONFIGURATION, one TcpIpIpVXCtrl container is created. Depending on the TcpIpDomainType, a TcpIpIpV4Ctrl or TcpIpIpV6Ctrl subcontainer is created.</p> <p>An ETHERNET-COMMUNICATION-CONNECTOR is associated with a NETWORK-ENDPOINT if there is a SOCKET-ADDRESS which references the NETWORK-ENDPOINT via SOCKET-ADDRESS/NETWORK-ENDPOINT-REF and which contains an APPLICATION-ENDPOINT that in turn references an ETHERNET-COMMUNICATION-CONNECTOR either via APPLICATION-ENDPOINT/CONNECTOR-REF or via APPLICATION-ENDPOINT/MULTICAST-CONNECTOR-REFS/MULTICAST-CONNECTOR-REF.</p>
TcpIpConfig/TcpIpCtrl/TcpIpIpVXCtrl/TcpIpIpV4Ctrl	<ul style="list-style-type: none"> <li>▶ TcpIpIpV4PathMtuEnabled is set to PATH-MTU-ENABLED</li> <li>▶ TcpIpIpV4PathMtuTimeout is set to PATH-MTU-TIMEOUT</li> </ul>
TcpIpConfig/TcpIpCtrl/TcpIpIpVXCtrl/TcpIpIpV6Ctrl	<ul style="list-style-type: none"> <li>▶ TcpIpIpV6PathMtuEnabled is set to IP-V-6-PATH-MTU-ENABLED</li> <li>▶ TcpIpIpV6PathMtuTimeout is set to IP-V-6-PATH-MTU-TIMEOUT</li> </ul>
TcpIpConfig/TcpIpIpConfig/TcpIpIpFragmentationConfig	<p>If at least one PDU that is sent or received via Udp needs fragmentation, i.e. its payload length plus the length of the Udp header plus the length of the IP header exceeds the MTU size defined in ETHERNET-COMMUNICATION-CONNECTOR/MAXIMUM-TRANSMISSION-UNIT, a container TcpIpIpConfig and a subcontainer TcpIpIpFragmentationConfig are created in TcpIpConfig. If any of the PDUs exceeding the MTU size is received, TcpIpIpFragmentationRxEnabled is set to true. If any of the PDUs exceeding the MTU size is sent, TcpIpIpFragmentationTxEnabled is set to OUTFORDER. The payload length of a PDU is calculated as described in <a href="#">Section 3.3.7, "PDU length calculation"</a>.</p>
TcpIpConfig/TcpIpTcpConfig	<p>If there is at least one SOCKET-ADDRESS element which represents a local IP port of the imported ECU-INSTANCE and which contains a TP-CONFIGURATION/TCP-TP element in its APPLICATION-ENDPOINT which in turn contains at least one of the parameters</p> <ul style="list-style-type: none"> <li>▶ KEEP-ALIVE-INTERVAL</li> <li>▶ KEEP-ALIVE-PROBES-MAX</li> </ul>

Configuration parameters	Mapping description
	<p>► KEEP-ALIVE-TIME</p> <p>or has its KEEP-ALIVE parameter set to <code>true</code> then the parameter <code>TcpIpTcpKeepAliveEnabled</code> is set to <code>true</code>.</p>
TcpIpConfig/TcpIpLocalAddr	<p>A NETWORK-ENDPOINT is considered to be configured at runtime in one of the following situations:</p> <ul style="list-style-type: none"> <li>► The IPV-4-CONFIGURATION of the NETWORK-ENDPOINT has its IPV-4-ADDRESS-SOURCE set to a value other than FIXED.</li> <li>► The IPV-6-CONFIGURATION of the NETWORK-ENDPOINT has its IPV-6-ADDRESS-SOURCE set to a value other than FIXED.</li> <li>► The NETWORK-ENDPOINT represents a multicast address that is configured at runtime.</li> </ul> <p>In all other cases, a NETWORK-ENDPOINT is considered to be configured at configuration time.</p> <p>A NETWORK-ENDPOINT represents a multicast address that is configured at runtime if either its IPV-4-ADDRESS or its IPV-6-ADDRESS is a multicast address and a <i>local</i> SOCKET-ADDRESS which meets the conditions described in <a href="#">Section 3.4.43.4, “Local SOCKET-ADDRESS elements configured by Sd”</a> references it via APPLICATION-ENDPOINT/NETWORK-ENDPOINT-REF.</p> <p>Every NETWORK-ENDPOINT that is associated with an ETHERNET-COMMUNICATION-CONNECTOR of a <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER is taken as input for the creation of <code>TcpIpLocalAddr</code> containers. For further information about ETHERNET-COMMUNICATION-CONTROLLER, see <a href="#">Section 3.4.17, “EthIf”</a>.</p> <p>If a NETWORK-ENDPOINT is considered to be configured at configuration time, one <code>TcpIpLocalAddr</code> is created for it. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the NETWORK-ENDPOINT.</p> <p>If a NETWORK-ENDPOINT is considered to be configured at runtime, one <code>TcpIpLocalAddr</code> container is created for each <i>local</i> SOCKET-ADDRESS that references it via its contained APPLICATION-ENDPOINT. The container name is <code>&lt;PREFIX&gt;&lt;name&gt;&lt;suffix&gt;</code>, where <code>&lt;name&gt;</code> is the SHORT-NAME of the NETWORK-ENDPOINT.</p> <p><code>TcpIpCtrlRef</code> references the <code>TcpIpCtrl</code> container created for the <i>virtual</i> ETHERNET-COMMUNICATION-CONTROLLER.</p>

Configuration parameters	Mapping description
	<p>The following parameters are configured using either the first IPV-4-CONFIGURATION which has IPV-4-ADDRESS-SOURCE set to FIXED or the first IPV-6-CONFIGURATION which has IPV-6-ADDRESS-SOURCE set to FIXED:</p> <p>TcpIpDomainType is set to TCPIP_AF_INET if NETWORK-ENDPOINT/NETWORK-ENDPOINT-ADDRESSES contains at least one IPV-4-CONFIGURATION entry and no IPV-6-CONFIGURATION entries. It is set to TCPIP_AF_INET6 if NETWORK-ENDPOINT/NETWORK-ENDPOINT-ADDRESSES contains at least one IPV-6-CONFIGURATION entry and no IPV-4-CONFIGURATION entries.</p> <p>TcpIpAddressType is set to TCPIP_MULTICAST if IPV-4-ADDRESS or IPV-6-ADDRESS represents a multicast address, otherwise TcpIpAddressType is set to TCPIP_UNICAST.</p>
<p>TcpIpConfig/TcpIpLocalAddr/TcpIpStaticIpAddressConfig</p>	<p>This container is only configured for NETWORK-ENDPOINT elements that do not represent multicast addresses that are configured at runtime.</p> <p>TcpIpStaticIpAddress is set to ANY if the NETWORK-ENDPOINT is a multicast address which is configured at runtime, or to IPV-4-ADDRESS, respectively IPV-6-ADDRESS in all other cases.</p> <p>If TcpIpAddressType is set to TCPIP_MULTICAST, TcpIpNetmask is not configured. Otherwise it is either set to the content of IPV-4-CONFIGURATION/NETWORK-MASK converted to the <i>Classless Inter-Domain Routing (CIDR)</i> notation, or to IPV-6-CONFIGURATION/IP-ADDRESS-PREFIX-LENGTH which is already provided in CIDR notation.</p> <p>If TcpIpAddressType is set to TCPIP_MULTICAST, TcpIpDefaultRouter is not configured. Otherwise the value of IPV-4-CONFIGURATION/DEFAULT-GATEWAY or IPV-6-CONFIGURATION/DEFAULT-ROUTER is retrieved. If that value does not equal ANY, it is used to configure TcpIpDefaultRouter.</p>
<p>TcpIpConfig/TcpIpLocalAddr/TcpIpAddrAssignment</p>	<p>For every element in the list of IPV-4-CONFIGURATION/IPV-4-ADDRESS-SOURCE entries respectively for every element in the list of IPV-6-CONFIGURATION/IPV-6-ADDRESS-SOURCE entries of the NETWORK-ENDPOINT a TcpIpAddrAssignment container is created. The container name is TcpIpAddrAssignment_&lt;assignmentMethod&gt;, where &lt;assignmentMethod&gt; depends on the value of IPV-4-ADDRESS-SOURCE or IPV-6-ADDRESS-SOURCE:</p> <ul style="list-style-type: none"> <li>▶ TCPIP_LINKLOCAL for AUTO-IP or LINK-LOCAL</li> <li>▶ TCPIP_LINKLOCAL_DOIP for AUTO-IP--DOIP or LINK-LOCAL--DOIP</li> <li>▶ TCPIP_DHCP for DHCPV-4 or DHCPV-6</li> </ul>

Configuration parameters	Mapping description
	<p>► TCPIP_STATIC for FIXED</p> <p>► TCPIP_IPV6_ROUTER for ROUTER-ADVERTISEMENT</p> <p>If IPV-4-ADDRESS-SOURCE is set to AUTO-IPDHCPV-4, two TcpIpAddrAssignment containers are created, one for AUTO-IP and one for DHCPV-4.</p> <p>TcpIpAssignmentMethod is set to &lt;assignmentMethod&gt;.</p> <p>If IPV-4-ADDRESS-SOURCE or IPV-6-ADDRESS-SOURCE contains a valid value other than AUTO-IPDHCPV-4, TcpIpAssignmentPriority is set to ASSIGNMENT-PRIORITY.</p> <p>TcpIpAssignmentTrigger is set to TCPIP_MANUAL if TcpIpStaticIpAddress is set to ANY.</p> <p>For NETWORK-ENDPOINT elements that do not contain any IPV-4-ADDRESS-SOURCE or any IPV-6-ADDRESS-SOURCE but represent a multicast address which is configured at runtime, one default TcpIpAddrAssignment container is created. Its name is TcpIpAddrAssignment_MANUAL. TcpIpAssignmentMethod is set to TCPIP_STATIC and TcpIpAssignmentTrigger is set to TCPIP_MANUAL.</p>
TcpIpConfig/TcpIpIpSecConfig	<p>If a NETWORK-ENDPOINT for which a TcpIpLocalAddr container was created contains one or more IP-SEC-RULE sub elements, one TcpIpIpSecConfig container is created.</p> <p>For each IP-SEC-RULE within a NETWORK-ENDPOINT, one TcpIpIpSecSecurityAssociationCfg container is created. Its name is SASS_&lt;name&gt;, where &lt;name&gt; is the short name of the IP-SEC-RULE. The following parameters are configured:</p> <p>TcpIpIpSecSecurityAssociationKeyExchangeMethod is set to DYNAMIC if IP-SEC-RULE/IKE-AUTHENTICATION-METHOD is available.</p> <p>TcpIpIpSecSecurityAssociationProtocol is set to IP-SEC-RULE/HEADER-TYPE if the value is either AH or ESP. For all other values, the parameter is not configured.</p> <p>For each NETWORK-ENDPOINT referenced by IP-SEC-RULE/REMOTE-IP-ADDRESS-REFS/REMOTE-IP-ADDRESS-REF, one TcpIpIpSecConnections container is created. Its name is SCON_&lt;name&gt;_&lt;remote_ip&gt;, where &lt;name&gt; is the short name of the IP-SEC-RULE and &lt;remote_ip&gt; is the short name of the remote NETWORK-ENDPOINT. If no NETWORK-ENDPOINT element is ref-</p>

Configuration parameters	Mapping description
	<p>erenced via IP-SEC-RULE/REMOTE-IP-ADDRESS-REFS/REMOTE-IP-ADDRESS-REF, one TcpIpIpSecConnections container is created. Its name is SCON_&lt;name&gt;. In both cases, the following parameters are configured:</p> <p>TcpIpIpSecDomainType is set to the same value as the TcpIpDomainType parameter of the TcpIpLocalAddr container which TcpIpIpSecLocalAddrRef refers to.</p> <p>TcpIpIpSecLocalAddrRef is configured to refer to the TcpIpLocalAddr container that has been created for the NETWORK-ENDPOINT of the IP-SEC-RULE.</p> <p>If the TcpIpIpSecConnections container has been created for a valid remote NETWORK-ENDPOINT, TcpIpIpSecRemoteAddrType is configured depending on whether it is representing a statically configured multicast address or not. The parameter is set to TCPIP_MULTICAST for statically configured multicast addresses, and to TCPIP_UNICAST for all other types of addresses. If the TcpIpIpSecConnections has not been created for a valid remote NETWORK-ENDPOINT, TcpIpIpSecRemoteAddrType is not configured.</p> <p>If the TcpIpIpSecConnections has been created for a valid remote NETWORK-ENDPOINT which represents a statically configured IP address, IpSecRemoteAddrConfig/SingleAddress/TcpIpIpSecRemoteAddr is configured to contain that static IP address.</p> <p>If IP-SEC-RULE/POLICY is set to IPSEC, TcpIpIpIpSecSecurityPolicy/TcpIpIpIpSecSecurityPolicyMechanism is configured as SECURED, if IP-SEC-RULE/POLICY is set to PASSTHROUGH, the parameter is configured as BYPASSED.</p> <p>If IP-SEC-RULE/DIRECTION is set to OUT, TcpIpIpIpSecSecurityPolicy/TcpIpIpIpSecSecurityPolicyDirection is configured as OUTBOUND, if IP-SEC-RULE/DIRECTION is set to IN, the parameter is configured as INBOUND.</p> <p>TcpIpIpIpSecSecurityPolicy/TcpIpIpIpSecSecurityPolicySecurityAssociationRef is configured to refer to the TcpIpIpSecSecurityAssociationCfg container that has been created for the IP-SEC-RULE.</p> <p>In the TcpIpIpIpSecSecurityPolicy container, two TcpIpIpIpSecSecurityRule sub containers are created, named TcpIpIpIpSecSecurityRule_LOCAL and TcpIpIpIpSecSecurityRule_REMOTE. The following parameters are configured:</p>



Configuration parameters	Mapping description
	<p>In <code>TcpIpIpIpSecSecurityRule_LOCAL</code>, <code>TcpIpIpSecSecurityPolicy-Direction</code> is set to <code>LOCAL</code>, in <code>TcpIpIpIpSecSecurityRule_REMOTE</code>, the parameter is set to <code>REMOTE</code>.</p> <p><code>TcpIpIpSecSecurityPolicyUpperLayer</code> is set to <code>IP-SEC-RULE/IP-PROTOCOL</code>.</p> <p>In <code>TcpIpIpIpSecSecurityRule_LOCAL</code>, <code>TcpIpIpSecSecurityPolicyStartPort</code> is set to <code>IP-SEC-RULE/LOCAL-PORT-RANGE-START</code>, in <code>TcpIpIpIpSecSecurityRule_REMOTE</code>, the parameter is set to <code>IP-SEC-RULE/REMOTE-PORT-RANGE-START</code>.</p> <p>In <code>TcpIpIpIpSecSecurityRule_LOCAL</code>, <code>TcpIpIpSecSecurityPolicyEndPort</code> is set to <code>IP-SEC-RULE/LOCAL-PORT-RANGE-END</code>, in <code>TcpIpIpIpSecSecurityRule_REMOTE</code>, the parameter is set to <code>IP-SEC-RULE/REMOTE-PORT-RANGE-END</code>.</p>

### 3.4.47. Tls

Configuration parameters	Mapping description
<code>Tls/TlsConnection</code>	<p>For every <code>TLS-CRYPTO-SERVICE-MAPPING</code> that is referenced by the <code>APPLICATION-ENDPOINT</code> of a local or remote <code>SOCKET-ADDRESS</code> of the configured <code>ECU-INSTANCE</code>, one <code>TlsConnection</code> container is created. Its name is set to <code>TLSConn_&lt;SocketAddressName&gt;</code>, where <code>&lt;SocketAddressName&gt;</code> is the <code>SHORT-NAME</code> of the <code>SOCKET-ADDRESS</code>.</p> <p><code>TlsCipherSuite</code> is configured depending on the first <code>TLS-CRYPTO-CIPHER-SUITE</code> element of the <code>TLS-CRYPTO-SERVICE-MAPPING</code>. <a href="#">Section 3.4.47.1, “TlsCipherSuite Mapping”</a> describes the mapping in detail.</p> <p><code>TlsEndpoint</code> is configured using <code>TLS-CRYPTO-SERVICE-MAPPING/CATEGORY</code> provided <code>TLS-CRYPTO-SERVICE-MAPPING/CATEGORY</code> either contains the value <code>TLS_CLIENT</code> or <code>TLS_SERVER</code>.</p> <p><code>TlsDtls</code> is configured depending on the protocol of the <code>SOCKET-ADDRESS</code>, which can be either <code>TCP</code> or <code>UDP</code>. The parameter is set to <code>false</code> for <code>TCP</code>, and to <code>true</code> for <code>UDP</code>.</p> <p>The references</p>

Configuration parameters	Mapping description
	<ul style="list-style-type: none"> <li>▶ CsmPreMasterSecretKeyRef</li> <li>▶ CsmMasterSecretKeyRef</li> <li>▶ CsmMACGenerateKeyRef</li> <li>▶ CsmMACVerifyKeyRef</li> <li>▶ CsmEncryptKeyRef</li> <li>▶ CsmDecryptKeyRef</li> <li>▶ CsmPRFPreMasterSecretJobRef</li> <li>▶ CsmPRFMasterSecretJobRef</li> <li>▶ CsmMACGenerateJobRef</li> <li>▶ CsmMACVerifyJobRef</li> <li>▶ CsmHashJobRef</li> <li>▶ CsmRandomJobRef</li> <li>▶ CsmEncryptJobRef</li> <li>▶ CsmDecryptJobRef</li> <li>▶ TlsCsmLocalCertSigGenerateJobRef</li> </ul> <p>are configured to refer to the corresponding containers in the Csm module configuration if a valid TlsCipherSuite was determined according to <a href="#">Section 3.4.47.1, “TlsCipherSuite Mapping”</a>. Please note that CsmEncryptKeyRef, CsmDecryptKeyRef, CsmEncryptJobRef, and CsmDecryptJobRef are not configured for cipher suites that do not provide encryption, e.g. TLS_PSK_WITH_NULL_SHA256.</p>

### 3.4.47.1. TlsCipherSuite Mapping

The TlsCipherSuite parameter is derived in one of two ways from the TLS-CRYPTO-SERVICE-MAPPING element. If the TLS-CRYPTO-SERVICE-MAPPING contains a TLS/cipherSuiteId SDG element which provides a valid TLS cipher suite ID, then the TlsCipherSuite is derived from that cipher suite ID. If no cipher suite id is available, TlsCipherSuite is derived from the properties of the CRYPTO-SERVICE-PRIMITIVE elements which the TLS-CRYPTO-SERVICE-MAPPING references via TLS-CRYPTO-SERVICE-MAPPING/KEY-EXCHANGE-REF, TLS-CRYPTO-SERVICE-MAPPING/AUTHENTICATION-REF, and TLS-CRYPTO-SERVICE-MAPPING/ENCRYPTION-REF.

The following table shows the configuration of TlsCipherSuite depending on the TLS/cipherSuiteId SDG element representing a cipher suite id.

Cipher suite Id	TlsCipherSuite
0x00B0	TLS_PSK_WITH_NULL_SHA256
0x00A8	TLS_PSK_WITH_AES_128_GCM_SHA256
0xC006	TLS_ECDHE_ECDSA_WITH_NULL_SHA
0xC02B	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

The following tables show the configuration of `TlsCipherSuite` depending on the properties of `TLS-CRYPTO-SERVICE-MAPPING/AUTHENTICATION-REF` and `TLS-CRYPTO-SERVICE-MAPPING/ENCRYPTION-REF`. Each of these `CRYPTO-SERVICE-PRIMITIVE` elements needs to contain a `TLS/keyOrHashLengthBytes` SDG, and the attributes `ALGORITHM-FAMILY`, and optionally `ALGORITHM-MODE`. For all `CRYPTO-SERVICE-PRIMITIVE` elements it is required that `ALGORITHM-SECONDARY-FAMILY` is not configured. Table entries that provide "n/a" in all of their fields indicate that the `TLS-CRYPTO-SERVICE-MAPPING` is expected not to refer to a `CRYPTO-SERVICE-PRIMITIVE` for the given reference. Moreover the `TlsCipherSuite` either needs to contain at least one element in `TLS-CRYPTO-SERVICE-MAPPING/KEY-EXCHANGE-REFS` if the represented cipher suite uses key exchange (indicated in the table by `KeyExchange: present`), or must not contain any element if the represented cipher suite does not use key exchange (indicated in the table by `KeyExchange: none`). In `TLS-CRYPTO-SERVICE-MAPPING/KEY-EXCHANGE-REFS` only the presence or absence of elements is checked, not their actual content.

`TLS-CRYPTO-SERVICE-MAPPING` for `TLS_PSK_WITH_NULL_SHA256`

CRYPTO-SERVICE-PRIMITIVE	ALGORITHM-FAMILY	ALGORITHM-MODE	KeyOrHashLengthBytes
KeyExchange: none			
Encryption	n/a	n/a	n/a
Authentication	CRYPTO_ALGO-FAM_SHA2_256	n/a	32

`TlsCryptoServiceMapping` for `TLS_PSK_WITH_AES_128_GCM_SHA256`

CRYPTO-SERVICE-PRIMITIVE	ALGORITHM-FAMILY	ALGORITHM-MODE	KeyOrHashLengthBytes
KeyExchange: none			
Encryption	CRYPTO_ALGO-FAM_AES	CRYPTO_ALGO-MODE_GCM	16
Authentication	CRYPTO_ALGO-FAM_SHA2_256	n/a	32

`TlsCryptoServiceMapping` for `TLS_ECDHE_ECDSA_WITH_NULL_SHA`

CRYPTO-SERVICE-PRIMITIVE	ALGORITHM-FAMILY	ALGORITHM-MODE	KeyOrHashLengthBytes
KeyExchange: present			
Encryption	n/a	n/a	n/a
Authentication	CRYPTO_ALGO-FAM_SHA1	n/a	20

TlsCryptoServiceMapping for TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256

CRYPTO-SERVICE-PRIMITIVE	ALGORITHM-FAMILY	ALGORITHM-MODE	KeyOrHashLengthBytes
KeyExchange: present			
Encryption	CRYPTO_ALGO-FAM_AES	CRYPTO_ALGO-MODE_GCM	16
Authentication	CRYPTO_ALGO-FAM_SHA2_256	n/a	32

### 3.4.48. UdpNm

Configuration parameters	Mapping description
UdpNmGlobalConfig	<p>UdpNmComUserDataSupport is set to true if UdpNmRxUserDataPduRef or UdpNmTxUserDataPduRef is set for any UdpNmChannelConfig or if any UDP-NM-CLUSTER linked to the imported ECU-INSTANCE has its NM-PNC-PARTICIPATION either not defined or set to true. Otherwise UdpNmComUserDataSupport is set to false.</p> <p>UdpNmPnResetTime is set to PN-RESET-TIME of the configured ECU-INSTANCE.</p> <p>The following parameters are set using the first NM-ECU of the imported ECU-INSTANCE:</p> <p>UdpNmUserDataEnabled is set to NM-USER-DATA-ENABLED.</p> <p>UdpNmRemoteSleepIndEnabled is set to NM-REMOTE-SLEEP-IND-ENABLED.</p> <p>UdpNmBusSynchronizationEnabled is set to NM-BUS-SYNCHRONIZATION-ENABLED.</p>

Configuration parameters	Mapping description
	<p>UdpNmStateChangeIndEnabled is set to NM-STATE-CHANGE-IND-ENABLED.</p> <p>UdpNmPassiveModeEnabled: see <a href="#">Section 3.4.4, “CanNm”</a>, CanNmPassiveModeEnabled.</p> <p>UdpNmPduRxIndicationEnabled is set to NM-PDU-RX-INDICATION-ENABLED.</p> <p>UdpNmComControlEnabled is set to NM-COM-CONTROL-ENABLED.</p> <p>UdpNmMainFunctionPeriod is set to NM-CYCLETIME-MAIN-FUNCTION.</p> <p>UdpNmComMNetworkHandleRef references the ComMChannel container that is created for the COMMUNICATION-CLUSTER referenced in COMMUNICATION-CLUSTER-REF.</p> <p>If inconsistencies are detected among parameters of multiple NM-ECU elements, a warning is reported.</p> <p>The following parameters are set using the first UDP-NM-CLUSTER-COUPLING of all UDP-NM-CLUSTER elements connected to the imported ECU-INSTANCE:</p> <p>If inconsistencies are detected among parameters of multiple UDP-NM-CLUSTER-COUPLING elements, a warning is reported.</p> <p>If a UDP-NM-CLUSTER configured as partial networking cluster (PNC) (see <a href="#">Section 3.4.14, “EcuC”</a>) belongs to the imported ECU-INSTANCE, the following parameters are set:</p> <p>UdpNmPnEiraRxNSduRef references the corresponding container in the EcuC module configuration.</p> <p>UdpNmPnEiraCalcEnabled is set to true.</p> <p>UdpNmPnInfo/UdpNmPnInfoOffset is set to PNC-VECTOR-OFFSET. If PNC-VECTOR-OFFSET is not defined, a warning is issued and UdpNmPnInfoOffset is not set.</p> <p>UdpNmPnInfo/UdpNmPnInfoLength is set to PNC-VECTOR-LENGTH. If PNC-VECTOR-LENGTH is not defined, a warning is issued and UdpNmPnInfoLength is not set.</p> <p>For the configuration of UdpNmPnFilterMaskByte all PNC-FILTER-DATA-MASK values of the ETHERNET-COMMUNICATION-CONNECTOR elements of</p>

Configuration parameters	Mapping description
	the configured ECU-INSTANCE are taken as input. The configuration algorithm is the same as described for CanNmPnFilterMaskByte in <a href="#">Section 3.4.4, “Can-Nm”</a> .
UdpNmChannelConfig	<p>For every ETHERNET-PHYSICAL-CHANNEL for which the following conditions hold, one UdpNmChannelConfig container is created.</p> <ul style="list-style-type: none"> <li>▶ The imported ECU-INSTANCE sends or receives at least one NM-PDU on the ETHERNET-PHYSICAL-CHANNEL.</li> <li>▶ A UDP-NM-CLUSTER which belongs to the imported ECU-INSTANCE references the ETHERNET-PHYSICAL-CHANNEL either directly via VLAN-REF or it references the ETHERNET-CLUSTER that contains the ETHERNET-PHYSICAL-CHANNEL.</li> </ul> <p>The container name is &lt;PREFIX&gt;&lt;name&gt;, where &lt;name&gt; is the SHORT-NAME of the ETHERNET-PHYSICAL-CHANNEL.</p> <p>A UDP-NM-CLUSTER belongs to the imported ECU-INSTANCE if at least one of its UDP-NM-NODE elements references an ETHERNET-COMMUNICATION-CONTROLLER of this ECU-INSTANCE.</p> <p>The NM-ECU used for configuring some of the UdpNmChannelConfig parameters is the NM-ECU which the first UDP-NM-NODE references via NM-IF-ECU-REF.</p> <p>UdpNmNodeDetectionEnabled is set to NM-CLUSTER/NM-NODE-DETECTION-ENABLED, or to NM-ECU/NM-NODE-DETECTION-ENABLED if NM-CLUSTER/NM-NODE-DETECTION-ENABLED is not available.</p> <p>UdpNmRepeatMsgIndEnabled is set to NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED, or to NM-ECU/NM-REPEAT-MSG-IND-ENABLED if NM-CLUSTER/NM-REPEAT-MSG-IND-ENABLED is not available.</p> <p>UdpNmNodeIdEnabled is set to NM-CLUSTER/NM-NODE-ID-ENABLED, or to NM-ECU/NM-NODE-ID-ENABLED if NM-CLUSTER/NM-NODE-ID-ENABLED is not available.</p> <p>UdpNmTimeoutTime is set to NM-NETWORK-TIMEOUT.</p> <p>UdpNmWaitBusSleepTime is set to NM-WAIT-BUS-SLEEP-TIME.</p> <p>UdpNmRepeatMessageTime is set to NM-REPEAT-MESSAGE-TIME.</p>

Configuration parameters	Mapping description
	<p><code>UdpNmRemoteSleepIndTime</code> is set to <code>NM-REMOTE-SLEEP-INDICATION-TIME</code>.</p> <p><code>UdpNmMsgCycleTime</code> is set to <code>NM-MSG-CYCLE-TIME</code>.</p> <p><code>UdpNmMsgTimeoutTime</code> is set to <code>NM-MESSAGE-TIMEOUT-TIME</code>.</p> <p><code>UdpNmImmediateNmCycleTime</code> is set to <code>NM-IMMEDIATE-NM-CYCLE-TIME</code>.</p> <p><code>UdpNmImmediateNmTransmissions</code> is set to <code>NM-IMMEDIATE-NM-TRANSMISSIONS</code>.</p> <p><code>UdpNmPduNidPosition</code> is set depending on the value of <code>NM-NID-POSITION</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>UDPNM_PDU_BYTE_0</code> for 0.</li> <li>▶ <code>UDPNM_PDU_BYTE_1</code> for 1.</li> <li>▶ <code>UDPNM_PDU_OFF</code> for any other value.</li> </ul> <p><code>UdpNmPduCbvPosition</code> is set depending on the value of <code>NM-CBV-POSITION</code>:</p> <ul style="list-style-type: none"> <li>▶ <code>UDPNM_PDU_BYTE_0</code> for 0.</li> <li>▶ <code>UDPNM_PDU_BYTE_1</code> for 1.</li> <li>▶ <code>UDPNM_PDU_OFF</code> for any other value.</li> </ul> <p><code>UdpNmRxPdu/&lt;EcuC container name&gt;/UdpNmRxPduRef</code> references the container in the <code>EcuC</code> module which has been created for the Rx NM-PDU. The Rx NM-PDU is the first NM-PDU that is transmitted on the <code>ETHERNET-PHYSICAL-CHANNEL</code> and that is referenced via <code>RX-NM-PDU-REFS/RX-NM-PDU-REF</code> of the <code>NM-NODE</code> associated with the <code>ETHERNET-PHYSICAL-CHANNEL</code>. The <code>NM-NODE</code> associated with an <code>ETHERNET-PHYSICAL-CHANNEL</code> is the first <code>NM-NODE</code> that receives an NM-PDU on the <code>ETHERNET-PHYSICAL-CHANNEL</code>. <code>&lt;EcuC container name&gt;</code> is the name of the referenced <code>EcuC</code> container. If no Rx NM-PDU is available, a warning is reported.</p> <p>If an <code>NmUserDataPdu</code> container has been created for the NM-PDU in the <code>EcuC</code> module configuration, <code>UdpNmUserDataRxPdu/UdpNmRxUserDataPduRef</code> references this container. The name of <code>UdpNmUserDataRxPdu</code> is set to the name of the referenced <code>EcuC</code> container.</p> <p><code>UdpNmTxPdu/UdpNmTxPduRef</code> references the container in the <code>EcuC</code> module which has been created for the Tx NM-PDU. The Tx NM-PDU is the first NM-PDU that is transmitted on the <code>ETHERNET-PHYSICAL-CHANNEL</code> and that is refer-</p>

Configuration parameters	Mapping description
	<p>enced via TX-NM-PDU-REFS/TX-NM-PDU-REF of the NM-NODE associated with the ETHERNET-PHYSICAL-CHANNEL. The name of UdpNmTxPdu is set to the name of the referenced EcuC container.</p> <p>If an NmUserDataPdu container has been created for the NM-PDU in the EcuC module configuration, UdpNmUserDataTxPdu/UdpNmTxUserDataPduRef references this container. The name of UdpNmUserDataTxPdu is set to the name of the referenced EcuC container.</p> <p>The following parameters are set using the first NM-NODE of the UDP-NM-CLUSTER connected to the imported ECU-INSTANCE. If inconsistencies are detected among parameters of multiple NM-NODE elements, a warning is reported.</p> <p>UdpNmMsgCycleOffset is set to NM-MSG-CYCLE-OFFSET.</p> <p>UdpNmNodeId is set to NM-NODE-ID.</p> <p>The configuration of PNC-related parameters is done in analogy to the configuration of the PNC-related CanNm parameters, see <a href="#">Section 3.4.4, “CanNm”</a>. The parameters and configuration containers obtain the prefix UdpNmPn. UdpNmPn-FilterMaskByte is not configured.</p>

### 3.4.49. Xcp

Configuration parameters	Mapping description
Xcp/XcpConfig/XcpPdu/XcpTxPdu	<p>For every GENERAL-PURPOSE-I-PDU that the imported ECU-INSTANCE sends without routing it (see <a href="#">Section 3.3.6, “PDU routing”</a>) and which has its CATEGORY field set to XCP, a XcpTxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the GENERAL-PURPOSE-I-PDU.</p> <p>XcpTxPdu/XcpTxPduRef is set to reference the configuration container of the GENERAL-PURPOSE-I-PDU in the EcuC module configuration.</p>
Xcp/XcpConfig/XcpPdu/XcpRxPdu	<p>For every GENERAL-PURPOSE-I-PDU that the imported ECU-INSTANCE receives without routing it (see <a href="#">Section 3.3.6, “PDU routing”</a>) and which has its CATEGORY field set to XCP, a XcpRxPdu container is created. The container name is &lt;PREFIX&gt;&lt;name&gt;&lt;INSTSUFFIX&gt;, where &lt;name&gt; is the SHORT-NAME of the GENERAL-PURPOSE-I-PDU.</p>





Configuration parameters	Mapping description
	XcpRxPdu/XcpRxPduRef is set to reference the configuration container of the GENERAL-PURPOSE-I-PDU in the EcuC module configuration.

## 4. Bibliography

### Bibliography

- [1] *AUTOSAR System Template*, Issue Version 4.2.0, Release 4.0, Revision 3, Publisher: AUTOSAR
- [2] *LIN Specification Package Revision 2.0*, Publish date: September 23, 2003, Publisher: LIN Consortium
- [3] *AUTOSAR System Template*, Issue Version 4.6.0, Release 4.2, Revision 2, Publisher: AUTOSAR
- [4] *AUTOSAR System Template*, Issue AUTOSAR CP Release 19-11, Publisher: AUTOSAR
- [5] *AUTOSAR System Template*, Issue AUTOSAR CP Release 4.4.0, Publisher: AUTOSAR
- [6] *AUTOSAR System Template*, Issue AUTOSAR CP Release 19-11, Publisher: AUTOSAR